



**MATEMATICKO-FYZIKÁLNÍ  
FAKULTA**  
Univerzita Karlova

## **BAKALÁŘSKÁ PRÁCE**

Jakub Saksa

### **Webový nástroj pro vizualizaci sekundární struktury RNA**

Katedra softwarového inženýrství

Vedoucí bakalářské práce: RNDr. David Hoksza, Ph.D.  
Studijní program: Informatika  
Studijní obor: Softwarové a datové inženýrství

Praha 2018

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V ..... dne..... Podpis autora

Název práce: Webový nástroj pro vizualizaci sekundární struktury RNA

Autor: Jakub Saksa

Katedra: Katedra softwarového inženýrství

Vedoucí bakalářské práce: RNDr. David Hoksza, Ph.D.

Abstrakt: Vizualizácia sekundárnej štruktúry RNA je otvoreným problémom, pretože neexistuje presný popis ako ju vykresliť, zobrazenie akceptované komunitou je zvykové. Prístup implementovaný v nástroji Traveler, ktorý vznikol na Katedre softwarového inženýrství MFF UK, používa pre vizualizáciu šablónu v podobe inej, už existujúcej štruktúry. Prvá verzia programu mala však niekoľko nedostatkov znemožňujúcich jeho použitie. Cieľom tejto práce bolo vylepšiť Traveler a vytvoriť tak nástroj, ktorého používanie bude pre užívateľa prístupnejšie. Vylepšeniami sú schopnosť spracovávať pseudouzly, medziformát pre obrázky sekundárnych štruktúr a úprava algoritmu pre tvorbu layoutu. Ďalším cieľom práce bolo vytvoriť webovú aplikáciu, ktorá bude slúžiť ako užívateľské rozhranie. Vytvorená aplikácia je schopná sama vybrať šablónu pre cieľovú štruktúru a vizualizovať ju.

Klíčová slova: bioinformatika, RNA, sekundárna štruktúra, vizualizácia

Title: Web-based tool for RNA secondary structure visualization

Author: Jakub Saksa

Department: Department of Software Engineering

Supervisor: RNDr. David Hoksza, Ph.D.

Abstract: Visualization of RNA secondary structure is an open problem because there does not exist a well-defined description of the layout process. The layout accepted by the community is habitual. The approach implemented by the Traveler tool, developed at Department of Software Engineering MFF UK, is based on using existing structure with known layout as a template for laying out structure with unknown layout. The first version of the program had few issues that made it difficult to use. The goal of the thesis was to improve Traveler so that the tool is more accessible. Improvements include ability to process pseudoknots, intermediate format for images of secondary structures and modification of the layout-creating algorithm. We created a web application that serves as an user interface for Traveler. The application is able to automatically choose a template for the structure too.

Keywords: bioinformatics, RNA, secondary structure, visualization

Rád by som poďakoval vedúcemu práce RNDr. Davidovi Hokszoovi, Ph.D. za množstvo času, ktoré mi venoval pri zodpovedaní otázok spojených s bakalárskou prácou, za jeho odborné vedenie, pripomienky, návrhy a trpezlivosť. V poslednom rade patrí vďaka mojej rodine a priateľom za ich podporu.



# Obsah

<b>Úvod.....</b>	<b>3</b>
<b>1 Úvod do RNA.....</b>	<b>4</b>
1.1 Čo je RNA.....	4
1.2 Funkcie RNA.....	4
1.3 Štruktúra nukleových kyselín.....	6
1.4 Sekundárna štruktúra RNA.....	7
<b>2 Vizualizácia sekundárnej štruktúry RNA.....</b>	<b>10</b>
2.1 Prístupy.....	10
2.2 Traveler.....	11
2.3 Algoritmus.....	11
2.4 Práca s programom.....	16
2.5 Nedostatky prvej verzie.....	17
2.6 Cieľ práce.....	19
<b>3 Webová aplikácia.....</b>	<b>21</b>
3.1 Frontend.....	21
3.2 Backend.....	21
3.2.1 Spracovanie dát.....	23
3.2.2 Výber šablóny.....	24
3.3 Používanie webovej aplikácie.....	25
3.3.1 Target.....	25
3.3.2 Template.....	25
3.3.3 Výsledok.....	26
3.3.4 Formáty súborov.....	27
3.4 Inštalácia.....	28
<b>4 Vylepšenia Traveleru.....</b>	<b>30</b>
4.1 Spracovávanie pseudouzlov.....	30
4.2 Medziformát pre obrázky.....	33
4.3 Algoritmus.....	34
<b>5 Experiment.....</b>	<b>39</b>
5.1 Popis experimentu.....	39
5.2 Dáta.....	39
5.3 Výsledky.....	39

<b>Záver.....</b>	<b>42</b>
<b>Zoznam použitej literatúry.....</b>	<b>43</b>
<b>Zoznam obrázkov.....</b>	<b>45</b>
<b>Zoznam tabuliek.....</b>	<b>46</b>

## Úvod

Ribonukleová kyselina (RNA) je jednou z kľúčových molekúl buniek. Štruktúra RNA je spätá s jej funkciou. O štruktúre je možné hovoriť na rôznych úrovniach. Primárnu štruktúru určuje sled nukleotidov, ktoré tvoria molekulu. Sekundárnu štruktúru tvorí zoznam párov nukleotidov (dva nukleotidy spojené väzbou). Terciárna štruktúra RNA popisuje tvar molekuly v priestore.

V súčasnej dobe existuje niekoľko nástrojov pre vizualizáciu sekundárnej štruktúry RNA. Vizualizácie, ktoré produkujú nie sú vždy prehľadné. Vedci zaoberajúci sa štúdiom RNA preferujú vykreslenie, v ktorom sa konzervované časti molekuly nachádzajú vždy na rovnakom mieste. Ako vykresliť sekundárnu štruktúru v podobe, ktorá je akceptovaná komunitou je stále otvorený problém. Nástroj Traveler tento problém rieši použitím algoritmu, ktorý vizualizuje cieľovú štruktúru na základe šablóny. Šablónu tvorí iná molekula RNA, ktorej vizualizácia už existuje. Vďaka tomu, že algoritmus funguje na základe porovnávania dvoch štruktúr, je možné vyznačiť, na ktorých miestach sa výsledok líši od šablóny.

Prvá verzia tohto programu má však niekoľko nedostatkov týkajúcich sa funkčnosti a používania. Traveler je prístupný len z príkazového riadku. Ďalším problémom je samotný algoritmus, ktorý zlyhá v prípade, že štruktúry sú lokálne príliš odlišné. Dôsledkom je vizualizácia, ktorá obsahuje veľké množstvo prekryvov a je neprehľadná. Traveler vo svojej súčasnej podobe pracuje s dvoma formátmi šablón (CRW a VARNA), čo obmedzuje množinu dát, s ktorou je schopný pracovať.

Cieľom tejto práce je odstrániť nedostatky prvej verzie Traveleru a vytvoriť tak nástroj, ktorý bude možné ľahko používať. Tento proces zahŕňa vytvorenie užívateľského rozhrania a doplnenie chýbajúcich funkcií.

# 1 Úvod do RNA

Táto práca sa zaoberá problematikou vizualizácie sekundárnej štruktúry RNA. Z tohto dôvodu je potrebné čitateľa najprv oboznámiť so základnými pojmami týkajúcimi sa RNA. Na nasledujúcich stranách vysvetlíme, čo RNA je, aké sú funkcie tejto molekuly, aký význam má sekundárna štruktúra a popíšeme spôsoby jej reprezentácie.

## 1.1 Čo je RNA

Ribonukleová kyselina, alebo RNA, je nukleová kyselina tvorená vláknom nukleotidov. Spolu s deoxyribonukleovou kyselinou (DNA) tvorí základ všetkého života. Často sa o nich hovorí ako o návodoch, ktoré obsahujú pokyny ku konštrukcii častí bunky.

Zásadný rozdiel medzi DNA a RNA je v štruktúre molekuly a množine nukleotidov, ktoré ju tvoria. Zatiaľ čo DNA sa skladá z dvoch komplementárnych vlákien nukleotidov, RNA tvorí jediné vlákno. Vlákna oboch kyselín sú zložené zo štyroch typov nukleotidových podjednotiek, ktoré sú vzájomne spojené vodíkovými väzbami. V prípade RNA ide o tieto podjednotky - adenín (A), cytozín (C), guanín (G) a uracyl (U). V DNA sa nevyskytuje uracyl, je nahradený tymínom (T). Ďalší rozdiel v zložení nájdeme v molekule päťuhlíkového cukru, na ktorý sa viažu dusíkaté bázy. V RNA sa vyskytuje ribóza, v DNA zase deoxyribóza. Tento rozdiel má za následok, menšiu stabilitu RNA, čo vedie na širšie spektrum funkcií, ktoré môže vykonávať (Alberts a kol., 2002).

Väzby medzi bázami nie sú náhodné. Vznikajú väzby, ktoré sú energeticky najvýhodnejšie. Tento spôsob párovania vedie väčšinou na vznik párov G-C, G-U a A-U v molekule RNA, sú však prípustné aj iné kombinácie. Pre RNA platí, že všetky dvojice nukleotidov sa nachádzajú na jedinom vlákne, teda RNA sa skladá sama na seba. V prípade DNA vznikajú páry G-C a A-T.

## 1.2 Funkcie RNA

RNA je všeobecne známa hlavne kvôli svojej funkcii pri prepise genetickej informácie z DNA. Okrem tejto funkcie plní niekoľko ďalších, nemenej dôležitých úloh, ktoré priblížime v nasledujúcich odsekoch.

Mediátorová RNA (mRNA, z angl. messenger RNA) bola prvý krát popísaná

v roku 1956. Prenáša informáciu z DNA do miesta syntézy bielkovín - ribozómu. Informácia v DNA nemôže byť dekódovaná priamo do proteínov, preto je najprv skopírovaná do mRNA, kde je kódovaná pomocou trojíc nukleotidov. Tento proces sa nazýva transkripcia. Úloha mRNA končí po jej presunutí do cytoplazmy, kde je pri procese translácie preložená pomocou ribozomálnej RNA.

Ribozomálna RNA (rRNA) je syntetizovaná v jadre bunky. Delí sa podľa veľkosti na malú a veľkú rRNA. Aspoň jedna malá a jedna veľká molekula rRNA je vždy súčasťou ribozómu. Tento ryp RNA sa vyznačuje svojou veľkosťou, rRNA je zo všetkých typov RNA najväčšia čo sa týka počtu nukleotidov. Vo veľkej ribozómovej podjednotke ich môže byť viac než 5000. Medzi organizmami, ktorých bunka má jednoznačne formované jadro (eukarioty) a organizmami, ktorým jadro chýba (prokarioty) je značný rozdiel v počte ribozómov. Eukariotické bunky obsahujú až 10 miliónov ribozómov, zatiaľ čo prokariotické bunky len 15000. Rôzne bunky v rámci jedného organizmu sa taktiež líšia v počte ribozómov, konkrétne číslo závisí od funkcie, ktorú bunka vykonáva. Vysoký počet ribozómov v bunke robí z rRNA najčastejšie sa vyskytujúci typ RNA (Alberts a kol., 2002).

Ďalšou funkciou RNA je prenos aminokyselín do ribozómov. Toto má na starosti transferová (prenosová) RNA (tRNA). Molekuly tRNA typicky pozostávajú z menej než 100 nukleotidov a skladajú sa do charakteristickej štvorlístkovej štruktúry. Pre každú z 20 aminokyselín existuje špecializovaná tRNA. Všetky tRNA majú podobnú štruktúru, pretože musia byť schopné pôsobiť na rovnakom mieste ribozómu.

Nakoniec spomenieme niekoľko menej známych typov RNA. Malá jaderná RNA (snRNA, z angl. small nuclear RNA) má za úlohu spracovávať pre-messenger RNA (hnRNA) v jadre bunky. Obsahuje v priemere 150 nukleotidov. Skupina molekúl snRNA označovaná ako snoRNA (small nucleolar RNA) zohráva dôležitú úlohu pri chemickej modifikácii rRNA. Reguláciu génovej expresie má na starosti molekula RNA pozostávajúca z približne 22 nukleotidov, nazývaná miRNA (microRNA). Tento typ RNA sa vyskytuje v bunkách zvierat, rastlín a niektorých vírusov. Boli však zaznamenané prípady, kedy sa miRNA vyskytla v mimobunkovom prostredí. Zaujímavosťou sú satelitné vírusy, ktoré dokážu replikovať svoju RNA v infikovaných bunkách, ide o špeciálny typ parazitickej RNA

(Alberts a kol., 2002).

### 1.3 Štruktúra nukleových kyselín

Štruktúra nukleových kyselín je často delená na štyri úrovne. Týmito úrovňami sú primárna, sekundárna, terciárna a kvartérna štruktúra.

Primárna štruktúra pozostáva z lineárnej postupnosti nukleotidov, ktoré tvoria kyselinu. Nukleotid sa skladá z dusíkatej bázy, päťuhlíkového cukru a fosfátových skupín. Primárna štruktúra nenesie informáciu o párovaní medzi nukleotidami v rámci molekuly.

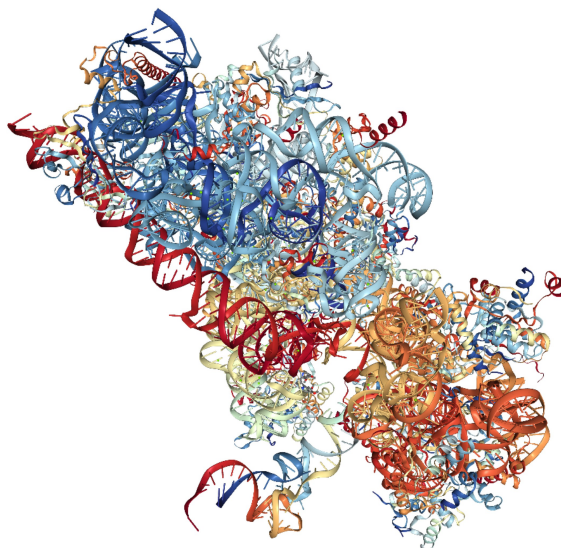
Sekundárna štruktúra (Obrázok 1.1) udáva dvojice nukleotidov, ktoré sú spojené vodíkovou väzbou. V prípade DNA sa spájajú nukleotidy z jedného vlákna s nukleotidmi nachádzajúcimi sa v druhom vlákne. U RNA vzniká párovanie medzi komplementárnymi časťami nukleotidového reťazca.



Obrázok 1.1: Príklad sekundárnej štruktúry RNA

Terciárna štruktúra popisuje molekulu v priestore. Zachytáva 3D súradnice atómov. Štruktúra tejto úrovne popisuje kyselinu najpresnejšie a zachytáva najviac informácií v porovnaní s ostatnými štruktúrami. Terciárna štruktúra je na prvý pohľad komplikovaná. Skladá sa však z menších útvarov, ktoré sú ľahko rozpoznateľné. Týmito útvarmi môžu byť napríklad dvojitý helix alebo triplex. Predikcia terciárnej štruktúry RNA je zložitý problém, preto nie je jednoduché štruktúru získať.

Kvartérna štruktúra nukleových kyselín popisuje ich interakcie s ostatnými molekulami. Môže ísť o dve rôzne molekuly nukleovej kyseliny alebo o nukleovú kyselinu a proteín.



Obrázok 1.2: Príklad terciárnej štruktúry RNA (PDB ID: 3J7A)

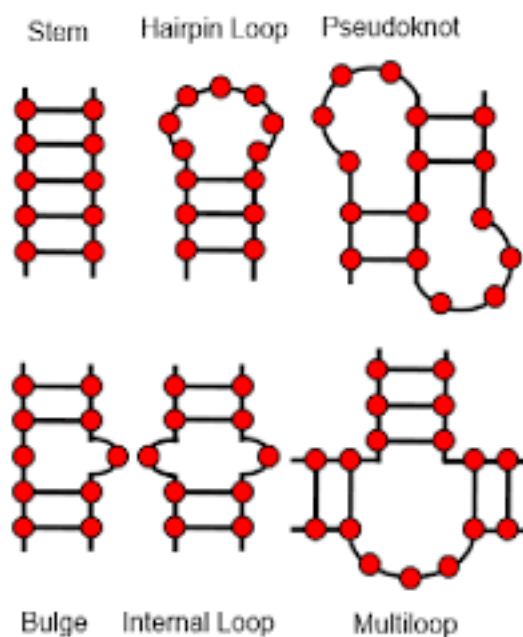
#### 1.4 Sekundárna štruktúra RNA

Ústrednou témou tejto práce je vizualizácia sekundárnej štruktúry RNA, preto v nasledujúcich riadkoch priblížime detaily tohto pohľadu na štruktúru.

Sekundárna štruktúra RNA udáva informáciu o párovaní nukleotidov. Môže byť reprezentovaná ako zoznam párov tvoriacich molekulu. V sekundárnej štruktúre nájdeme ďalší rozdiel medzi RNA a DNA. Jediné vlákno RNA často vytvára komplexné motívy, čo je zapríčinené jej zvýšenou schopnosťou vytvárať vodíkové väzby. DNA naopak väčšinou existuje ako úplne spárovaná dvojité špirála (double helix). Okrem informácie o pároch nukleotidov vieme zo sekundárnej štruktúry zistiť, ktoré časti molekuly sa v priestore nachádzajú blízko pri sebe. Tieto časti zodpovedajú oblastiam v sekundárnej štruktúre.

Sekundárna štruktúra je rozdelená do rôznych motívov. Najvšeobecnejšími motívmi sú špirála (helix) a slučka (loop). Špirála je postupnosť párov, slučka je skupina nespárovaných nukleotidov, ktorá je obklopená špirálami. Kombinácie týchto motívov vytvárajú ďalšie kategórie. Príkladmi zložitejších motívov sú

stem loop, branch loop alebo pseudouzol (pseudoknot). Stem loop je slučka s jedinou špirálou, branch loop je slučka s aspoň troma špirálami (slučka s dvoma špirálami vytvára internal loop). Pseudouzol je motív sekundárnej štruktúry, ktorý vzniká z dvoch stem-loop motívov, kde polovica jednej špirály je vložená medzi obe polovice druhej špirály.



Obrázok 1.3: Motívy sekundárnej štruktúry RNA (Chheda a Gupta, 2014)

Existuje niekoľko spôsobov ako uchovávať sekundárnu štruktúru. Tieto techniky neslúžia k vizualizácii sekundárnej štruktúry ale len k uloženiu informácie, ktorú sekundárna štruktúra nesie. Najznámejšími metódami sú grafová reprezentácia, mountain representation a stromová reprezentácia.

•**Grafová reprezentácia** sekundárnej štruktúry je graf, ktorého vrcholmi sú jednotlivé nukleotidy a hrany reprezentujú väzby medzi nimi (Hofacker a Stadler). Graf musí spĺňať tieto podmienky:

1. Každý vrchol sa smie vyskytnúť v najviac jednom páre.
2. Vrcholy, ktoré sú spárované musia byť oddelené aspoň troma nespárovanými vrcholmi.
3. Žiadne dva páry  $(i,j)$  a  $(k,l)$  sa nepretínajú, t.j. nikdy nenastane  $i < k < j < l$ .



- **Mountain representation** nahliada na štruktúru ako na dvojdimenzionálny graf (nie graf ako ho poznáme v informatike). V tomto grafe je x-ová súradnica  $k$  pozícia nukleotidu v sekvencii a y-ová súradnica je číslo  $m(k)$ , ktoré udáva počet párov obklopujúcich tento nukleotid. Táto reprezentácia je vhodná pre veľké štruktúry.
- **Stromová reprezentácia** využíva tvz. “dot-bracket notation”. Celá štruktúra je reprezentovaná ako jediný reťazec znakov “.”, “(” a “)”. Pre každý pár  $i$  a  $j$ , ( $i < j$ ) je v reťazci znak “(” na pozícii  $i$  a “)” na pozícii  $j$ . Nespárované nukleotidy sú znázornené bodkou (“.”). K takto skonštruovanému reťazcu sa pridáva reťazec s označeniami nukleotidov v sekvencii.

## 2 Vizualizácia sekundárnej štruktúry RNA

### 2.1 Prístupy

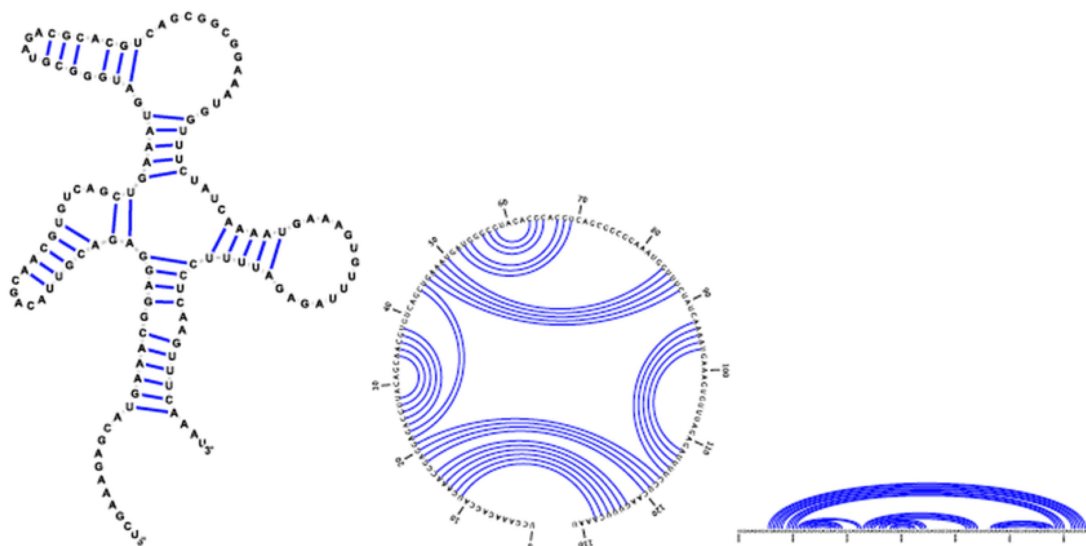
Existuje niekoľko nástrojov pre vizualizáciu layoutu sekundárnej štruktúry RNA, ktoré sa líšia prístupom k problému. V súčasnosti väčšina nástrojov používa jeden z nasledujúcich spôsobov zobrazenia molekuly RNA: lineárne rozmiestnenie, kruhové rozmiestnenie alebo rovinný graf.

Kresba štruktúry pomocou lineárneho rozmiestnenia zobrazuje párovanie medzi bázami ako oblúky nakreslené nad alebo pod čiarou, na ktorej je lineárne zobrazená RNA sekvencia (Obrázok 2.1, vpravo). Výhodou tejto metódy je jednoduché rozpoznanie špirál, ktoré sú reprezentované sériou zanorených oblúkov. Nevýhodou je množstvo priestoru zabraného oblúkmi reprezentujúcimi párovanie. Dĺžka oblúku rastie priamo úmerne so vzdialenosťou medzi bázami, ktoré spája. Tento fakt spôsobuje, že plocha obrázku rastie kvadraticky vzhľadom k dĺžke sekvencie (Ponty a Leclerc, 2015). Navyše nukleotidy tvoriace pár sú na obrázku ďaleko od seba a neaproximujú terciárnu štruktúru.

Modifikovanou verziou lineárneho rozmiestnenia je kruhové rozmiestnenie. Tento spôsob vizualizácie zobrazuje RNA sekvenciu ako kružnicu (Obrázok 2.1, stred). Párovanie je zobrazené buď ako oblúky vo vnútri kružnice alebo ako tetivy spájajúce jednotlivé páry. Kruhové rozmiestnenie odstraňuje nedostatok svojho lineárneho náprotivku týkajúci sa množstva plochy obrázku. To má však za následok, že dĺžka oblúku presne nezodpovedá vzdialenosti báz, čo zase vedie k chybám pri rozoznávaní špirál. Špirály sú v kruhovom rozmiestnení znázornené ako oblúky alebo tetivy v tesnej blízkosti kružnice a môžu byť mylne označené za lokálne motívy (Ponty a Leclerc, 2015).

Nástroje, ktoré používajú pre reprezentáciu layoutu sekundárnej štruktúry rovinné grafy (Obrázok 2.1, vľavo) sa snažia štruktúru zobrazit' ako schému terciárnej štruktúry. Špirály sú spravidla zobrazené ako postupnosť párov, nukleotidy v rámci páru sú spojené rovnou čiarou. Okrem tejto podobnosti sa prístupy rôznych nástrojov líšia. Výhodou tejto metódy sú prehľadné obrázky, na ktorých sa dajú jednoducho rozoznať jednotlivé motívy. Medzi nástroje používajúce túto metódu patrí aj objekt záujmu tejto práce - Traveler (Eliáš a Hoksza, 2016). Inými príkladmi

sú nástroje RNAView (Yang a kol., 2003), VARNA (Darty a kol., 2009), RNAPlot (Lorenz a kol., 2011), PseudoViewer (Byun a Han, 2006) alebo jViz.RNA (Wiese a kol., 2005).



Obrázok 2.1: Príklady zobrazenia sekundárnej štruktúry RNA, zľava: rovinný graf, kruhové rozmiestnenie a lineárne rozmiestnenie

## 2.2 Traveler

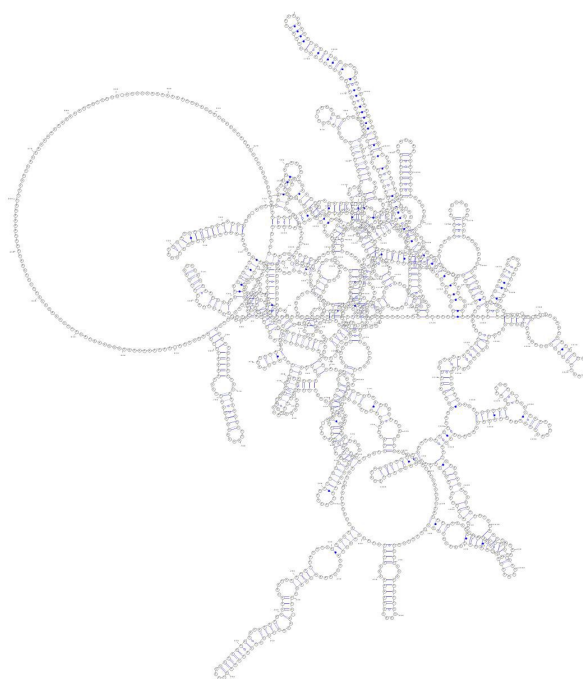
Nástroj Traveler (Eliáš a Hoksza, 2016) vznikol na Katedre softwarového inžinýrství MFF UK. Motiváciou pre jeho vznik bola neexistencia nástrojov, ktoré sú schopné rozumne vizualizovať veľké RNA molekuly. V týchto prípadoch je dôležité aby bolo zobrazenie sekundárnej štruktúry prehľadné a aby obsahovalo funkčne dôležité motívy na konkrétnych miestach v priestore, čo sa nástrojom často nedarí dosiahnuť, pretože existuje nekonečne mnoho spôsobov ako nakresliť tú istú sekundárnu štruktúru. Vedci upredostňujú layout v podobe rovinného grafu, toto zobrazenie je považované komunitou za správne. Tento efekt Traveler dosahuje tým, že sekundárnu štruktúru vizualizuje pomocou šablóny. Šablóna je iná molekula RNA, ktorej sekundárna štruktúra už bola rozumne vizualizovaná.

## 2.3 Algoritmus

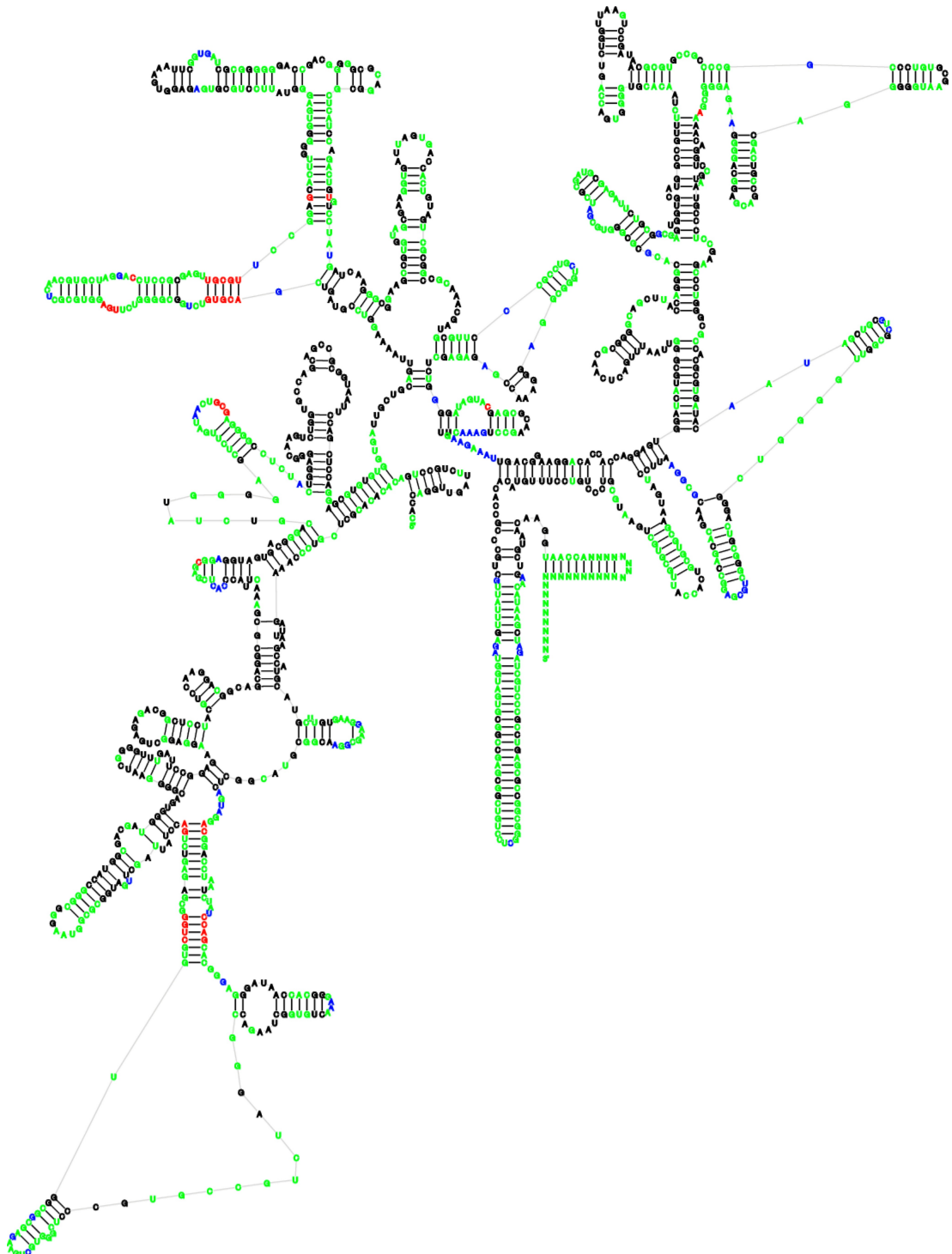
Traveler pracuje so stromovou reprezentáciou vstupných štruktúr. Sekundárnu štruktúru RNA, zapísanú v dot-bracket formáte je možné previesť na

zakorenený strom nasledujúcim spôsobom. Založíme strom, ktorý obsahuje jediný vrchol (koreň). Potrebujeme si pamätať, v ktorom vrchole sa momentálne nachádzame, skôr ako prečítame prvý znak sa nachádzame v koreni. Prechádzame reťazec, obsahujúci znaky “(”, “)” a “.”, zľava doprava. Vždy keď narazíme na znak “(”, pridáme súčasnému vrcholu nového syna a zostúpime do neho. Pri prečítaní znaku “)” vystúpime zo súčasného vrcholu do jeho rodiča. Pre znak “.” pridáme vrcholu nového syna ale nebudeme do neho zostupovať. Okrem vyššie spomínaných znakov môže popis sekundárnej štruktúry obsahovať aj znaky “[” a “{” (resp. ľavé zátvorky). Tieto znaky reprezentujú pseudouzly. Traveler pseudouzly ignoruje a tieto znaky nahradí znakom “.”.

Po úspešnom prevedení vstupných štruktúr na stromy program spočíta stromovú editačnú vzdialenosť (TED - Tree Edit Distance) medzi nimi. Algoritmov zaoberajúcich sa TED je niekoľko. Traveler implementuje RTED (Pawlik a Augsten, 2011). Poznamenajme, že pôvodný algoritmus pre TED (Tai, 1979) má zložitosť  $O(m^3n^3)$ , RTED tento výsledok zlepšuje na  $O(m^3)$  a potrebuje  $O(mn)$  pamäte.



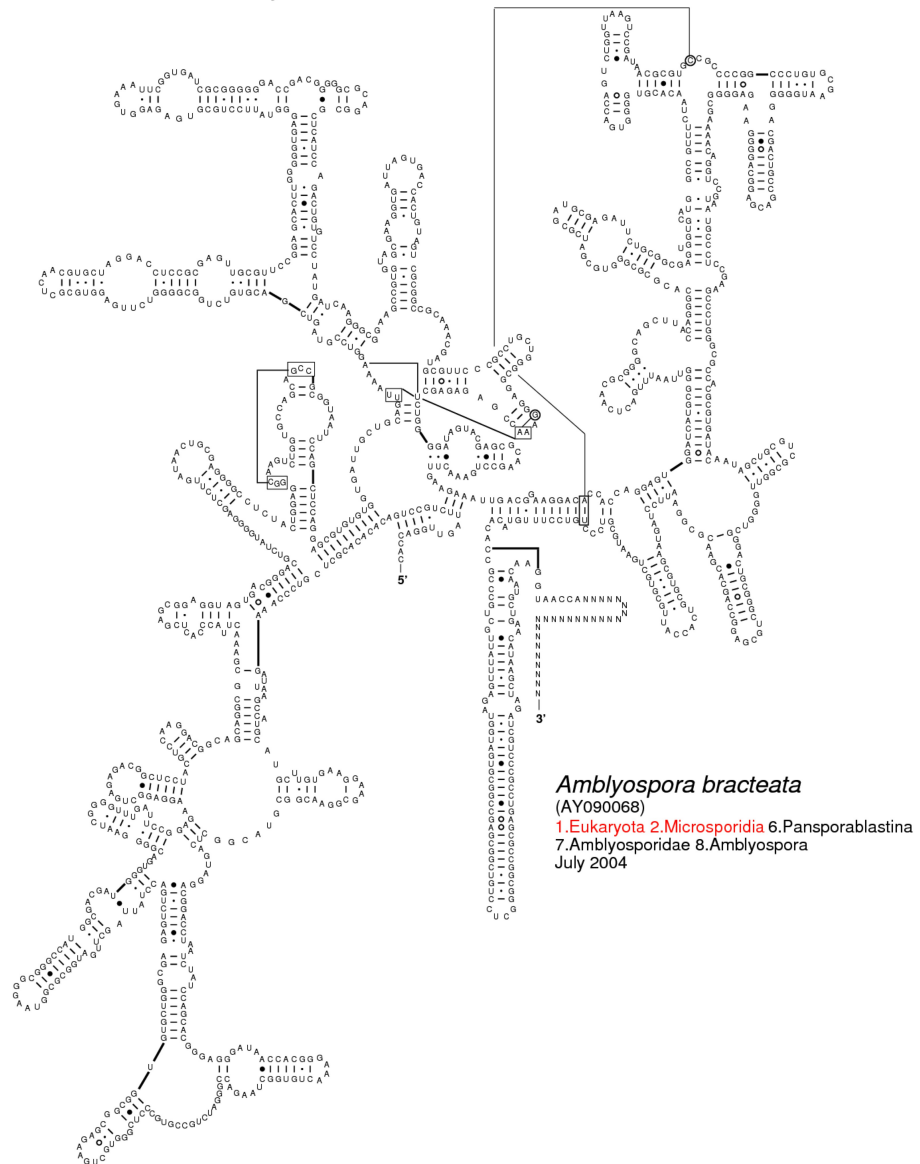
Obrázok 2.2: Vizualizácia sekundárnej štruktúry RNA človeka pomocou nástroja Varna (Darty a kol., 2009)



Obrázok 2.3: Vizualizácia sekundárnej štruktúry *Amblyospora bracteata* 18s rRNA pomocou nástroja Traveler (použitá šablóna: 18S human rRNA)

Výsledkom výpočtu TED sú dva údaje. Jedným z dvojice výstupných údajov je číslo udávajúce rozdielnosť stromov, druhým je minimálna postupnosť operácií, pomocou ktorých je možné premeniť vizualizovanú štruktúru na šablónu. Používané operácie sú vloženie vrcholu, zmazanie vrcholu a premenovanie vrcholu. V algoritmoch pre

### Secondary Structure: small subunit ribosomal RNA

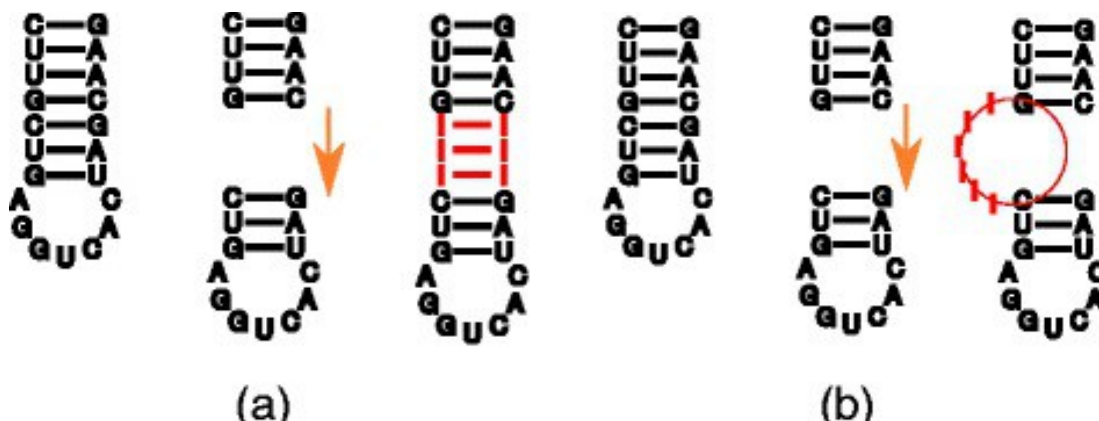


Citation and related information available at <http://www.rna.icmb.utexas.edu>

Obrázok 2.4: Očakávané zobrazenie sekundárnej štruktúry *Amblyospora bracteata* 18s rRNA (Cannone a kol, 2002)

TED je povolené aby mali operácie rôzne váhy, Traveler používa rovnakú váhu pre všetky operácie.

Ďalším krokom je premena štruktúry na šablónu postupným aplikovaním operácií, ktoré sme získali výpočtom TED. Pri vykonávaní zmien je potrebné dbať na efekt týchto zmien na šablónu. Odobrať vrchol zo stromu znamená odobrať jeden pár nukleotidov zo šablóny a to má za následok voľné miesto vo výslednej vizualizácii. Rovnako pridávanie vrcholu do stromu sa prejaví ako nový pár nukleotidov na obrázku. Traveler preto prepočítava súradnice vrcholov pri ich pridávaní resp. mazaní. Rozlišuje dva prípady: vnútorný vrchol a list. Vnútorný vrchol reprezentuje pár nukleotidov nachádzajúci sa na stonke. V tomto prípade je potrebné rozdeliť podštruktúru na časť pred vkladánym resp. mazaným párom a časť po ňom. V závislosti na operácii Traveler prepočíta súradnice ovplyvnených vrcholov, teda pri vkladaní nového páru posunie časť stonky aby vytvoril miesto pre pár a pri mazaní zmení súradnice tak, aby bolo zaplnené voľné miesto po zmazanom páre. Ak operácia narába s listom, táto situácia reprezentuje vloženie jedného nukleotidu do slučky. V tomto prípade však môže nastať, že slučka ešte neexistuje a je potrebné ju vytvoriť. Situáciu nie je zložité vyriešiť a postupuje sa podobne ako pri vkladaní páru do stonky. Namiesto páru sa ale vloží na jednu stranu stonky slučka obsahujúca jediný nukleotid. Ak už slučka existuje, nový nukleotid je do nej vložený po prepočítaní súradníc ostatných nukleotidov na slučke. Pri mazaní je postup analogický (Eliáš a Hoksza, 2016).



Obrázok 2.5: Úprava šablóny pri vkladaní nukleotidov (Eliáš a Hoksza, 2016)

Špeciálnym prípadom, ktorý sa vyskytuje pri premene štruktúry na šablónu, sú tzv. multibranches. Ide o podštruktúry pozostávajúce z niekoľkých stoniek nachádzajúcich sa na slučke. V stromovej reprezentácii štruktúry sa tento typ podštruktúry prejaví ako vrchol s aspoň dvoma synmi, ktorí nie sú listy. Ak by boli

zmeny v pozíciách párov v tomto prípade riešené rovnako ako pri zmenách v stonkách alebo slučkách, dochádzalo by k veľkému zásahu do šablóny a teda výraznej zmene rozloženia vo výslednej vizualizácii. Preto ak je to možné, tak sa pozície párov nemenia. Ak je ale vložených alebo zmazaných veľké množstvo párov je potrebné zmeniť pozície, čo musí byť propagované do všetkých stoniek tvoriacich multibranch (Eliáš a Hoksza, 2016).

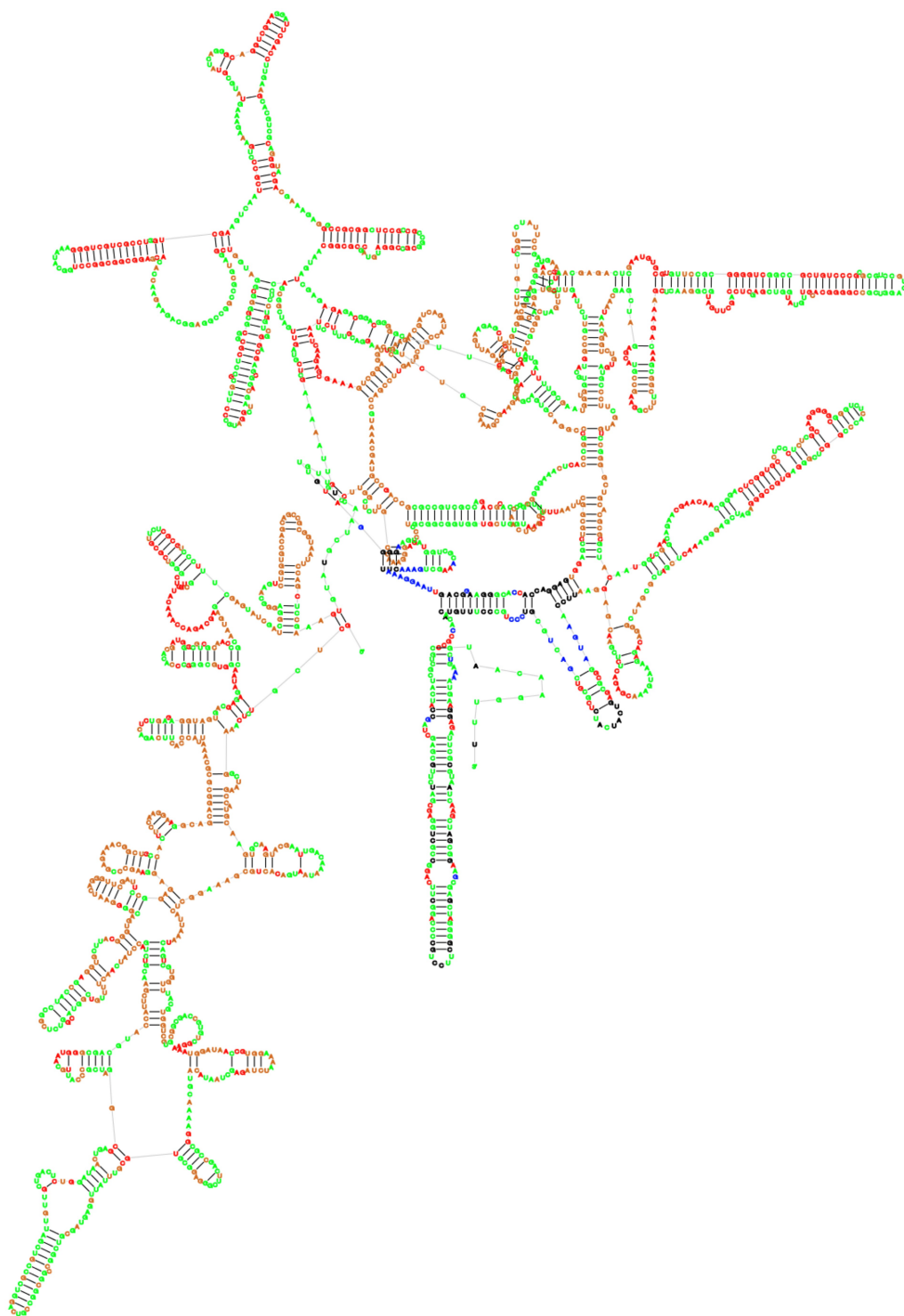
Po upravení vstupnej štruktúry nasleduje postprocessing. V rámci tejto fázy sa upravuje rozloženie nukleotidov tak, aby bola vizualizácia čo najkvalitnejšia. To zahŕňa napríklad zarovnávanie stoniek (páry tvoriace jednu stonku by mali ležať na jednej priamke) alebo minimalizovanie počtu prekryvov. Prekryvy sú odstránené tak, že pre každý podstrom, ktorého vrcholy kolidujú s vrcholmi iného podstromu program vyskúša niekoľko iných pozícií a vyberie tú, ktorá má za následok najmenej prekryvov.

## **2.4 Práca s programom**

Traveler je nutné ovládať pomocou príkazového riadku. Vstupné súbory sú programu odovzdané v podobe argumentov a pomocou prepínačov je možné vybrať požadovanú funkciu. Program pracuje s popisom sekundárnej štruktúry vo Fasta formáte. Fasta súbor obsahuje na prvom riadku názov štruktúry, ktorú popisuje a na ďalších riadkoch štruktúru zapísanú pomocou reťazca so skratkami nukleotidov a reťazca popisujúceho párovanie nukleotidov v dot-bracket formáte. Pre šablónu je potrebné ešte pridať súbor popisujúci rozloženie nukleotidov - obrázok. Prvá verzia programu pracuje s obrázkami vo formátoch CRW (.ps) a VARNA (.svg). Výstup programu je obrázok v oboch formátoch.

Okrem vizualizácie štruktúry, Traveler ponúka možnosť vygenerovať súbor obsahujúci vzdialenosť medzi vstupnou štruktúrou a šablónou a postupnosť operácií potrebných na prevedenie štruktúry na šablónu. Tento súbor je potom schopný prijať na vstupe a vygenerovať na jeho základe vizualizáciu. Podrobnú ukážku a pokyny pre prácu s programom je možné nájsť na adrese [github.com/davidhoksza/traveler](https://github.com/davidhoksza/traveler).





Obrázok 2.6: Príklad problému s mapovaním

## 2.5 Nedostatky prvej verzie

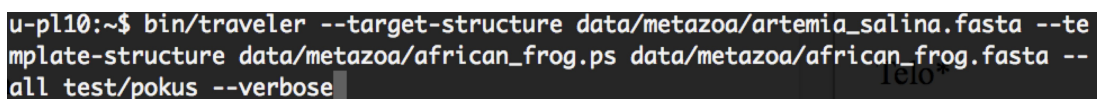
Prvá verzia nástroja Traveler, ktorá vznikla v roku 2016, má niekoľko nedostatkov. Tieto nedostatky sa týkajú ako funkčnosti programu, tak aj užívateľskej prívetivosti a práce s programom.

Ako bolo spomenuté vyššie program používa na vizualizáciu sekundárnej štruktúry RNA šablónu. Pomocou tejto šablóny sa snaží vizualizovať cieľovú štruktúru na základe TED medzi stromovými reprezentáciami oboch štruktúr. Algoritmus funguje bez problémov v prípadoch, kedy sú cieľová štruktúra a šablóna dostatočne podobné, t.j. na prevedenie cieľovej štruktúry na šablónu nie je potrebné veľké množstvo operácií (insert, delete, rename). Problém nastáva v prípade, že štruktúry sú príliš odlišné. Ak je editačná vzdialenosť medzi stromami reprezentujúcimi štruktúry príliš veľká, má to za následok to, že mapovanie (postupnosť editačných operácií), ktoré Traveler zvolí je nesprávne. Nesprávne znamená, že vizualizácia štruktúry na základe tohto mapovania nie je korektná. Vyskytuje sa v nej veľké množstvo prekryvov a podštruktúry nie sú vykreslené správne. Podstatou problému je, že algoritmus, ktorý Traveler používa, sa snaží nájsť postupnosť operácií s najnižšou cenou globálne. To ale nebude fungovať v prípade, že štruktúry sú takmer identické ale výrazne sa líšia v jednej oblasti. Tento jav nastáva ak v jednej štruktúre chýba párovanie v rámci niektorej podštruktúry, pretože ešte nebolo zaznamenané a v druhej štruktúre už zaznamenané je. Príkladom takejto situácie je vizualizácia sekundárnej štruktúry RNA ryby s použitím ľudskej RNA ako šablóny (Obrázok 2.6). Algoritmus sa snaží minimalizovať prevod štruktúry na šablónu a v rámci toho vyprodukuje mapovanie, ktoré vedie na zlé zobrazenie štruktúry.

Ďalším nedostatkom sú formáty obrázkov reprezentujúcich šablóny. V prvej verzii Traveler podporoval dva formáty - CRW (PostScript) a VARNA (SVG). CRW (Comparative RNA Web (Cannone a kol., 2002)) sa zaoberá skúmaním RNA štruktúr a obsahuje databázu obrázkov sekundárnych štruktúr v Gutellovom formáte. Nepříjemnosťou je, že to zďaleka nie sú všetky typy súborov, v ktorých môže byť zachytené zobrazenie sekundárnej štruktúry a ak aj obrázok je súbor s koncovkou .ps alebo .svg, Traveler mu nebude rozumieť v prípade, že nie je zapísaný podľa pravidiel formátov, ktoré pozná. Ak užívateľ nepracuje s nástrojom, ktorý dokáže vyprodukovať obrázok sekundárnej štruktúry RNA vo formáte CRW alebo VARNA, resp. nemá k dispozícii štruktúru v jednom z týchto formátov, ktorú môže použiť ako šablónu, nebude môcť Traveler použiť. Navyše oba podporované formáty, nesú okrem podstatnej informácie o pozíciách nukleotidov množstvo ďalších informácií

(hlavička, definície, ...). Tieto informácie Traveler nepotrebuje ale sú potrebné kvôli správne mu zobrazeniu súboru. Preto prevod do podporovaných formátov nie je triviálny a obyčajnému užívateľovi môže robiť problémy.

Program je možné používať v rámci príkazového riadku. Pri spustení programu je potrebné pomocou prepínačov odovzdať programu vstupné a výstupné súbory. To nie je pre užívateľa Traveleru pohodlné, pretože pre úspešné spustenie programu je nutné zadať aspoň tri prepínače (cieľová štruktúra, šablóna, úloha) a štyri názvy súborov (popis štruktúr, obrázok šablóny a výstupný súbor). Ak sa užívateľ rozhodne použiť rozvinuté názvy prepínačov, ktoré sú oproti ich skráteným verziám prehľadnejšie, môže sa stať že príkaz na spustenie programu zaberá niekoľko riadkov (pri šírke konzoly 80 znakov). Keďže výstupom Traveleru môže byť (a často je) obrázok bolo by vhodné tento obrázok prezentovať užívateľovi hneď ako je hotový. To vedie na nejakú formu grafického užívateľského rozhrania, ktorá by odstránila všetky spomenuté nepríjemnosti so spúšťaním programu a prezentovaním jeho výsledkov.



```
u-pl10:~$ bin/traveler --target-structure data/metazoa/artemia_salina.fasta --template-structure data/metazoa/african_frog.ps data/metazoa/african_frog.fasta --all test/pokus --verbose
```

Obrázok 2.7: Príklad spustenia programu

Program vo svojej pôvodnej podobe nedokázal spracovávať pseudouzly. Pri parsovaní vstupných štruktúr zahodil informáciu o tomto motíve a ďalej pracuje so sekundárnou štruktúrou bez pseudouzlov. Pseudouzly sú však validnou súčasťou sekundárnej štruktúry RNA a mali by byť zaznamenané vo výsledku.

## 2.6 Cieľ práce

Cieľom tejto práce bolo odstrániť vyššie spomenuté nedostatky nástroja Traveler, prípadne doplniť chýbajúce funkčnosti.

Čo sa týka fungovania programu, hlavný bodom záujmu bolo odstránenie alebo zmiernenie problému s algoritmom, ktorý nastáva, keď je cieľová štruktúra príliš odlišná od šablóny. Okrem toho bola doplnená funkcionálna spracovávanie pseudouzlov. Zamerali sme sa tiež na problémy s používaním programu, konkrétne jeho zložité ovládanie a množinu vstupných súborov, ktoré je schopný spracovať.

Úprava algoritmu spočívala v rozdelení štruktúr na menšie časti, venujeme sa

jej spolu s implementáciou spracovávania pseudouzlov a tvorbou medziformátu v kapitole 4. Problém s ovládaním programu sme vyriešili vytvorením webovej aplikácie (kapitola 3).

### 3 Webová aplikácia

Jedným z cieľov práce bolo vytvoriť grafické užívateľské rozhranie, ktoré umožní jednoduché používanie nástroja Traveler. GUI vo forme webovej aplikácie má výhodu v tom, že nie je závislé na platforme a program je možné používať okamžite bez predchádzajúcej inštalácie a konfigurácie.

Traveler GUI je rozdelené na dve časti - frontend a backend. Frontend beží vo webovom prehliadači užívateľa a slúži na komunikáciu s programom a prezentovanie výsledkov. Backend beží na vzdialenom počítači (server), ktorý spúšťa samotný Traveler a odovzdáva mu vstup zadaný užívateľom. V tejto kapitole bližšie popíšeme implementáciu webovej aplikácie a ako s ňou pracovať.

#### 3.1 Frontend

Užívateľ komunikuje s Travelerom pomocou frontendu. Ide o zmes HTML, CSS a JavaScript kódu. Formulár, ktorý užívateľ vyplňa vstupnými dátami je naimplementovaný pomocou HTML, rovnako ako členenie stránky na oblasti (menu, formulár, ...). CSS štýly sme využili pri úprave vzhľadu aplikácie. Pri implementácii frontendu sme taktiež použili Bootstrap (Bootstrap 4.1, 2018). Táto knižnica má široké využitie pri tvorbe layoutu webových stránok a zjednodušuje používanie CSS štýlov. JavaScript našiel uplatnenie pri spracovávaní udalostí (kliknutie užívateľa) a komunikácií s backendom (odosielanie vstupných dát a reakcia na výsledok). Pre prepojenie frontendu a backendu bola použitá technológia AJAX.

Frontend webovej aplikácie umožňuje zadávať popis sekundárnej štruktúry RNA, ktorú chce užívateľ vizualizovať. Ďalej poskytuje nápovedu k jednotlivým komponentám, s ktorými môže užívateľ manipulovať a kontroluje, či vstup obsahuje všetko potrebné pre korektný priebeh vizualizácie (užívateľská dokumentácia je obsiahnutá v kapitole 3.3).

#### 3.2 Backend

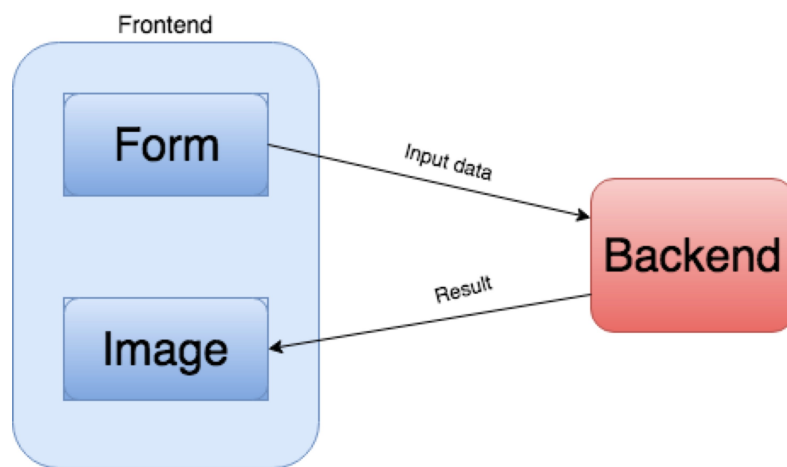
Táto časť webovej aplikácie má na starosti spracovanie všetkých dát, ktoré užívateľ zadal pomocou frontendu.

Jej úlohami sú:

- predspracovanie dát od užívateľa,
- spúšťanie všetkých potrebných programov (vrátane nástroja Traveler),

- odovzdávanie výsledkov frontendu a propagácia chýb, ktoré nastali pri výpočte.

Voľba operačného systému pre náš server bola jednoznačná, pretože Traveler je dostupný len pre unixové systémy. Konkrétna verzia nie je dôležitá, my používame CentOS. Ako webový server sme zvolili Apache. Ide o v súčasnosti najrozšírenejší server s najväčšou podporou. Ostatné alternatívy ako napríklad Nginx alebo Node.js by nepriniesli žiadnu (pre nás dôležitú) výhodu. Najväčšia časť funkcionality backendu je naimplementovaná v jazyku PHP. Ide o script, ktorému frontend posiela vstupné dáta. Okrem tohto scriptu backend využíva shell script na jednoduché predspracovanie vstupu. PHP disponuje širokou paletou funkcií pre prácu so súborami (vytváranie dočasných súborov, prechádzanie súborovým systémom, ...) a umožňuje jednoduché spúšťanie programov z PHP scriptu. Obe funkcionality sú často využívané pri spracovávaní užívateľských dát backendom. Spúšťanie programov je potrebné pri volaní Traveleru a shell scriptov. Prechádzanie

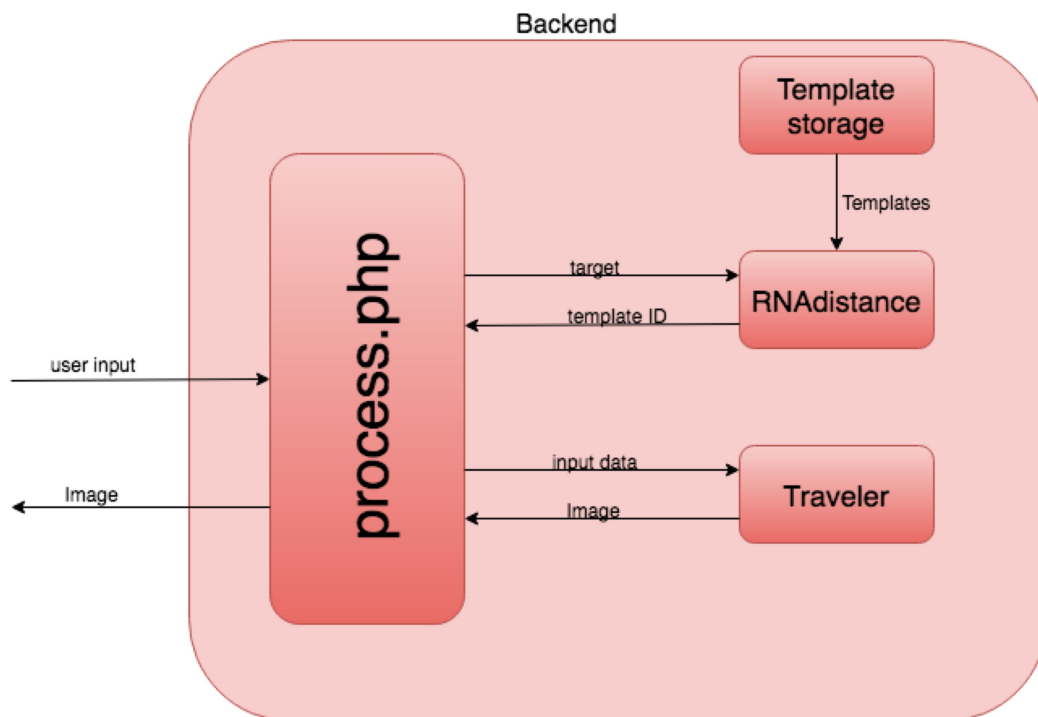


Obrázok 3.1: Frontend

súborovým systémom je použité pri vyberaní šablóny a výbere výstupného súboru vygenerovaného Travelerom (Traveler generuje niekoľko výstupných súborov). Ďalším dôvodom pre použitie PHP je možnosť odstrániť duplicitný HTML kód, rozložením spoločných častí do samostatných PHP súborov. Server poskytuje webovému prehliadaču užívateľa vždy len jednu stránku, ktorej obsah sa mení v závislosti od GET parametrov.

Stránka je zložená z troch častí:

- **header.php** obsahuje HTML kód s hlavičkou súboru a odkazmi na používané zdroje,
- **telo** závisí od GET parametrov (formulár alebo nápoveda),
- **footer.php** obsahuje HTML kód s uzatváracími tagmi.



Obrázok 3.2: Backend

### 3.2.1 Spracovanie dát

Užívateľ vyplní formulár vstupnými dátami, na základe ktorých chce získať vizualizáciu sekundárnej štruktúry konkrétnej RNA molekuly. Tento formulár je následne odoslaný na server. Komunikácia frontendu a backendu je asynchrónna. To znamená, že pri obdržaní odpovede od servru nedochádza k obnoveniu stránky ani k presmerovaniu na inú stránku. Zároveň je webový prehliadač užívateľa schopný vykonávať nejakú činnosť namiesto čakania na odpoveď. Pre tento spôsob komunikácie medzi klientom a serverom sme sa rozhodli, pretože vizualizácia veľkých RNA molekúl (rozsah v tisíckach nukleotidov) môže trvať až niekoľko minút. Počas čakania na výsledok je užívateľovi zobrazená ikona načítavania spolu s upozornením, že vybavenie jeho žiadosti môže trvať netriviálne dlho.

Po obdržaní dát začne backend s ich spracovaním. V prípade, že užívateľ zadal spolu so štruktúrou, ktorú chce vizualizovať aj šablónu, PHP script na strane serveru spustí Traveler a čaká na výsledok. Ak program dobehol v poriadku a neohlásil žiadnu chybu, backend odovzdá výsledok v podobe SVG obrázku frontendu, ktorý ho zobrazí užívateľovi. Ak pri výpočte nastala chyba (neočakávaný formát súboru, nekorektný vstup, ...) je propagovaná do klienta a zobrazená užívateľovi. Užívateľ má okrem ručného zadávania šablóny možnosť nechať server zvoliť šablónu. V tejto situácii predchádza spusteniu Traveleru výber vhodnej šablóny. Rozhodovanie o tom, ktorú šablónu je vhodné použiť má na starosti nástroj RNAdistance z balíka ViennaRNA. ViennaRNA pozostáva z niekoľkých nástrojov na analýzu sekundárnych štruktúr RNA. Ako súčasť backendu používame len jeden z nich, vyššie spomenutý RNAdistance. Výstupom tohto programu je číslo udávajúce podobnosť dvoch sekundárnych štruktúr. Tento nástroj sa rovnako ako Traveler pozerá na RNA molekuly ako na stromy a o ich podobnosti rozhoduje pomocou stromovej editačnej vzdialenosti. Súčasťou backendu je niekoľko predpripravených štruktúr, ktoré môžu byť použité ako šablóny. RNAdistance postupne porovná všetky z nich so štruktúrou, ktorú zadal užívateľ. PHP script si ukladá výsledky aby bol schopný vybrať šablónu s najlepším skóre. Ďalej prebieha spracovanie vstupu od užívateľa rovnako ako v prípade, keď je šablóna explicitne zadaná.

Okrem možnosti zobrazit' výsledok v podobe SVG obrázku sme uvažovali, že do webového GUI zakomponujeme ďalšiu funkcionálnosť Traveleru - mapovanie medzi stromovými reprezentáciami sekundárnych štruktúr. Od tejto možnosti sme nakoniec upustili, pretože formát súboru s mapovaním je špecifický pre Traveler a vizualizácia na základe mapovania vytvára rovnaké obrázky ako bežné používanie nástroja.

### **3.2.2 Výber šablóny**

Aby bol Traveler schopný vyprodukovať kvalitné vykreslenie sekundárnej štruktúry, potrebuje aby bola šablóna dostatočne podobná cieľovej RNA. Na podobnosť dvoch molekúl sa môžeme pozeráť z hľadiska homológie. V biológii znamená homológia štruktúr, že majú spoločný pôvod. Platí, že spoločný pôvod v rámci evolúcie vedie na vysokú mieru podobnosti medzi dvoma štruktúrami



(Koonin, 2005). Pre výber šablóny ale nepotrebujeme kontext v podobe homológie, stačí sa na molekuly RNA pozerat' ako na stromy. Keďže sekundárnu štruktúru každej RNA vieme previesť na strom, môžeme podobnosť vyjadriť ako stromovú editačnú vzdialenosť. Experimentálne sme overili, že takýto prístup vyberá správnu šablónu. Za správnu šablónu považujeme takú šablónu, ktorú by sme zvolili na základe homológie. Experiment a jeho výsledky sa nachádzajú v kapitole 5.

### 3.3 Používanie webovej aplikácie

Webová aplikácia slúži ako grafické užívateľské rozhranie pre nástroj Traveler, sídli na adrese [traveler.projekty.ms.mff.cuni.cz](http://traveler.projekty.ms.mff.cuni.cz). Poskytuje užívateľovi možnosť jednoduchej práce s programom a pridáva možnosť vizualizovať štruktúru bez poskytnutia šablóny (tá bude zvolená serverom). Pre používanie webového GUI nie je potrebné nič okrem webového prehliadača. Po načítaní stránky sa užívateľovi zobrazí formulár s dvoma sekciami: Target a Template.

#### 3.3.1 Target

Sekcia formulára označená ako Target (Obrázok 3.3, horná časť) slúži na špecifikáciu sekundárnej štruktúry RNA, ktorú chce užívateľ vizualizovať. To môže vykonať dvoma spôsobmi. Ak má užívateľ k dispozícii súbor s popisom štruktúry vo formáte FASTA, môže ho použiť kliknutím na tlačítko "Fasta". V prípade, že užívateľ nemá prístup k takémuto súboru, stačí aby zápis štruktúry vo FASTA formáte vložil do textového poľa. Popis FASTA formátu, spolu so všetkými ostatnými formátmi súborov, ktoré môžu byť použité ako vstup pre Traveler sa nachádza na konci kapitoly.

#### 3.3.2 Template

Táto sekcia zhromažďuje informácie o šablóne (Obrázok 3.3, dolná časť). V prípade, že užívateľ nechce špecifikovať šablónu alebo ju nemá k dispozícii, môže zakliknúť checkbox s popisom "Let Traveler choose template". Ak je táto možnosť zvolená, zvyšok sekcie Template nie je viditeľný. Pre explicitné zadanie šablóny sú potrebné dva súbory. Prvým súborom je popis sekundárnej štruktúry. Rovnako ako v sekcii Target je vyžadovaný FASTA formát a je možné nahrať súbor alebo vložiť

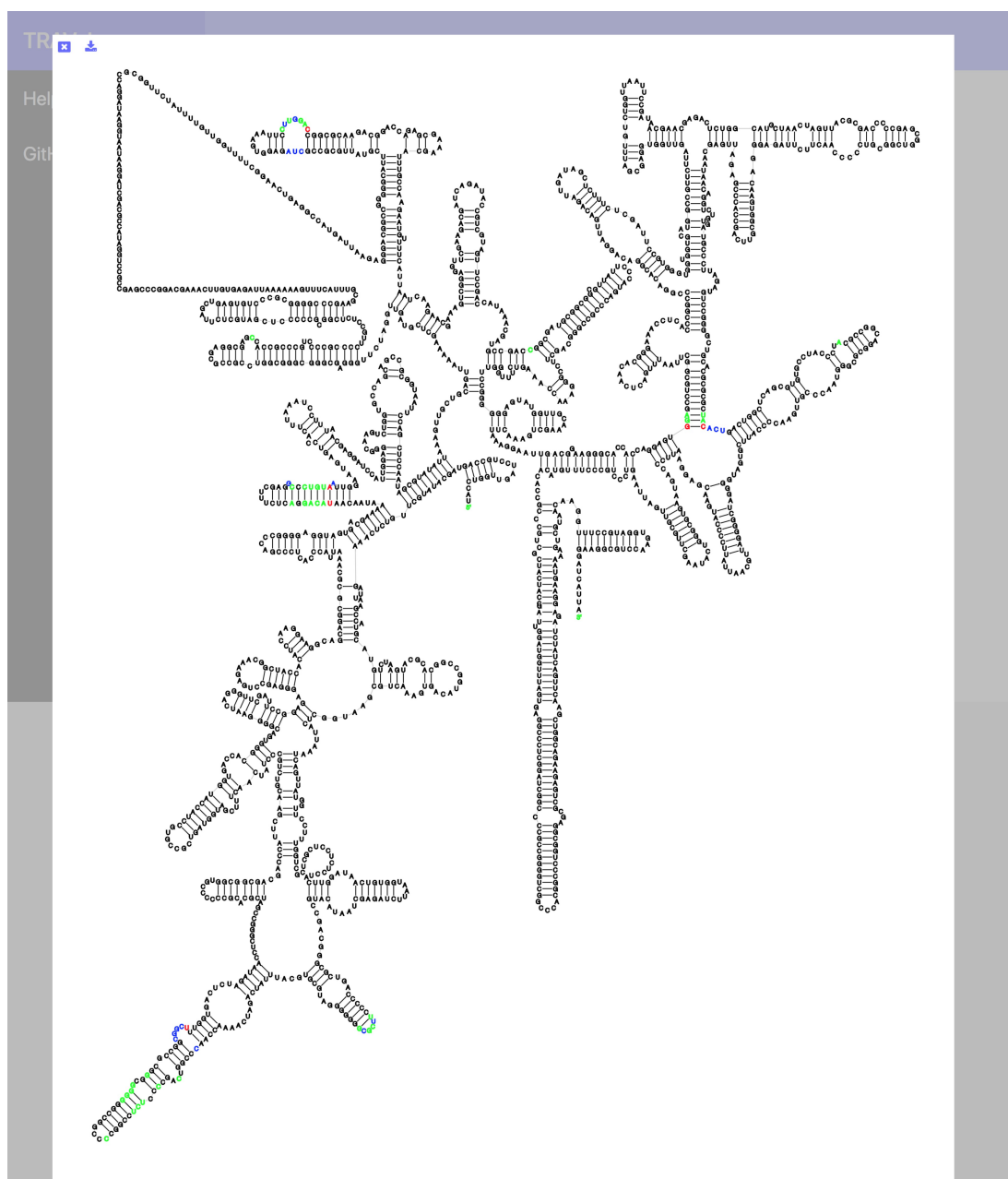
popis do textového poľa. Nasleduje popis rozloženia nukleotidov v podobe obrázku. Užívateľ musí špecifikovať formát súboru (vyberie jednu z možností) a potom nahrať súbor v tomto formáte pomocou tlačítka “Image”. Traveler podporuje formáty SVG, PostScript a XML. Poslednou možnosťou v tejto sekcii je checkbox “Overlaps”. Jeho zakliknutím užívateľ hovorí Traveleru, že chce aby vo výslednej vizualizácii vyznačil prekryvy.

The screenshot shows the TRAVeLer web application interface. On the left is a dark sidebar with 'Help' and 'GitHub' links. The main area has a light blue header with the 'TRAVeLer' logo and a hamburger menu. Below the header, the 'Target' section contains a 'Fasta' button and a text input field. The 'Template' section includes a checkbox 'Let Traveler choose template', another 'Fasta' button and text input field, radio buttons for 'SVG', 'PostScript', and 'Traveler (XML)', an 'Image' button, and an 'Overlaps' checkbox. A 'Run' button is at the bottom right.

Obrázok 3.3: Formulár pre zadávanie vstupných dát

### 3.3.3 Výsledok

Po vyplnení formulára užívateľ klikne na tlačítko “Run”. Ak je formulár vyplnený správne, spustí sa výpočet a užívateľovi je zobrazené upozornenie, že vizualizácia môže trvať niekoľko minút. V prípade, že niektorá časť formulára ostala nevyplnená, užívateľovi je zobrazená výzva aby chybu odstránil. Po ukončení výpočtu sa buď zobrazí chybová hláška (ak došlo k chybe na strane serveru) alebo obrázok s vizualizáciou sekundárnej štruktúry. V ľavom hornom rohu obrázku sa nachádzajú tlačítka, ktoré umožňujú obrázok zahodiť alebo uložiť.



Obrázok 3.4: Prezentácia výsledku pomocou webovej aplikácie

### 3.3.4 Formáty súborov

Program sám o sebe generuje obrázky v troch formátoch (SVG, PS, XML). Webová aplikácia prijíma obrázky na vstupe, rovnako ako program, vo všetkých formátoch ale výstupný obrázok je dostupný len vo formáte SVG. Ďalej popíšeme formáty súborov, s ktorými dokáže Traveler pracovať.

- **FASTA** formát slúži na popis párovania nukleotidov v sekundárnej štruktúre

RNA. Dokument v tomto formáte je obyčajný textový dokument, ktorého prvý riadok začína znakom “>”, za ktorým nasleduje popis molekuly. Na ďalších riadkoch nasleduje sekvencia nukleotidov reprezentovaných ich skratkami (A,C,G,U). Poslednou časťou FASTA súboru je popis párovania nukleotidov v dot-bracket formáte. Dĺžka jedného riadku by nemala presiahnuť 120 znakov, zvyčajne majú riadky dĺžku 80 znakov. Tento formát Traveler používa len pre vstupné súbory.

- **SVG** je značkovací jazyk z rodiny XML. Primárne je určený na popis obrázkov dvojrozmernej vektorovej grafiky. Traveler tento formát používa rovnako pre vstupné súbory (šablóna), ako aj pre výstupné obrázky. V prípade, že je ako šablóna použitý obrázok, ktorý nebol vygenerovaný nástrojom Traveler ale pochádza z externého zdroja (napríklad nástroj Varna), je potrebné pred jeho použitím odstrániť prebytočný text. Traveler identifikuje tag popisujúci text ako nukleotid a pri výpočte dojde k chybe. V obrázku by sa mali nachádzať len znaky reprezentujúce nukleotidy.
- **PostScript** je ďalším z Travelerom podporovaných formátov, ale len v prípade, že ide o obrázok z CRW. Rovnako ako SVG formát je možné .ps súbory použiť ako šablónu a súčasne vizualizácia sekundárnej štruktúry vytvorená programom je dostupná v tomto formáte.
- **XML** reprezentácia obrázku sekundárnej štruktúry vznikla v rámci tejto práce. Je riešením ako odstrániť obmedzenie Traveleru na dva konkrétne formáty (SVG a PostScript). XML súbory sú prijímané programom a rovnako je schopný ich generovať. Detailný popis štruktúry XML súborov je možné nájsť v kapitole 4.2.

### 3.4 Inštalácia

1. Nainštalujte webový server Apache.
2. Nainštalujte PHP.

3. Stiahnite zdrojové kódy webovej aplikácie pomocou príkazu:  
`git clone https://github.com/JakubSaksa/traveler_web`

4. Pomocou nasledujúcich príkazov spustíte script, ktorý skopíruje zdrojové kódy webovej aplikácie do priečinka, kde ich Apache očakáva a nainštaluje Traveler. Script má jeden argument, cestu k cieľovému priečinku (pre CentOS “/var/www/html”):  
`cd traveler_web`  
`./config.sh /var/www/html`

5. Stiahnite ViennaRNA balíček pomocou príkazu:  
`wget`  
`https://www.tbi.univie.ac.at/RNA/download/sourcecode/2_4_x/ViennaRNA-2.4.6.tar.gz`

Rozbaľte stiahnutý archív:

```
tar xvzf ViennaRNA-2.4.6
```

Nainštalujete potrebné programy:

```
cd ViennaRNA-2.4.6
./configure
make
make install
```

Ak sa balíček nepodarilo nainštalovať, skúste postupovať podľa pokynov v súbore “INSTALL”, ktorý je pribalený k distribúcii ViennaRNA. Podrobné pokyny pre inštaláciu môžete nájsť tiež tu: <https://www.tbi.univie.ac.at/RNA/tutorial/>

6. Uistite sa, že Apache má dostatočné práva na spúšťanie Traveleru a programov z balíka ViennaRNA.

## 4 Vylepšenia Traveleru

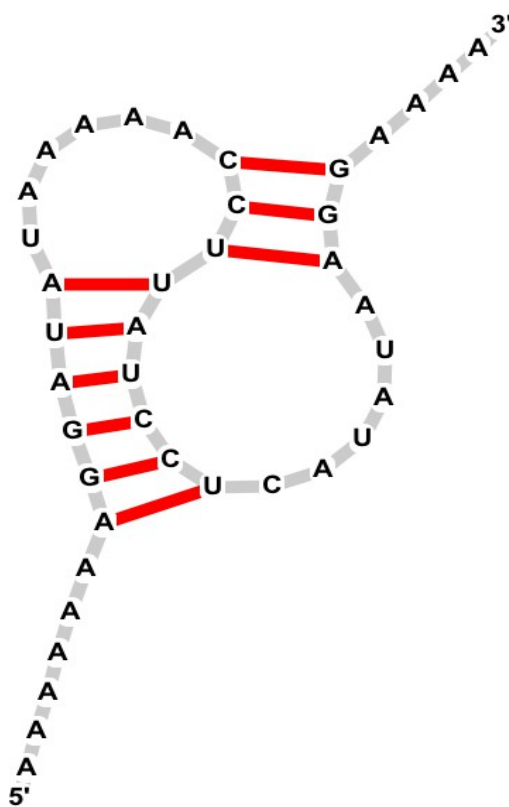
Okrem vytvorenia webovej aplikácie, ktorá bude slúžiť ako užívateľské rozhranie pre Traveler, bolo cieľom práce odstrániť nedostatky prvej verzie programu. Chýbajúce funkcionality programu majú rôzny charakter:

- Ignorovanie pseudouzlov pri vizualizácii sekundárnej štruktúry znižuje správnosť výstupu.
- Problém s formátmi šablón obmedzuje veľkosť množiny dát, s ktorými je program schopný pracovať.
- Algoritmus používaný pri mapovaní cieľovej štruktúry na šablónu.

### 4.1 Spracovávanie pseudouzlov

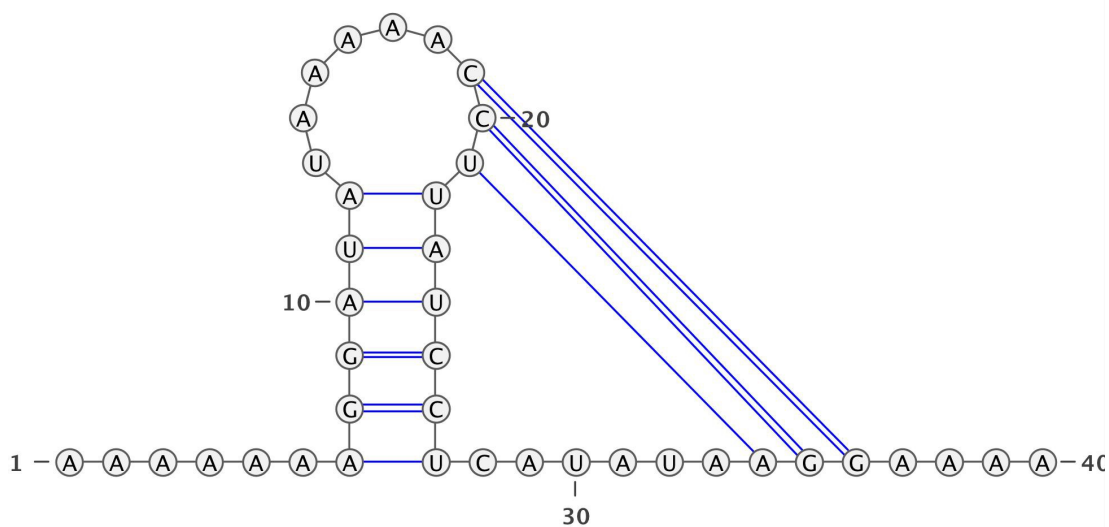
Pseudouzol je jeden z motívov, ktoré sa môžu vyskytnúť v sekundárnej štruktúre RNA. Od klasických motívov, ako napríklad slučka alebo špirála sa líši tým, že na zaznamenanie párovania nukleotidov vrámci štruktúry, ktorá obsahuje pseudouzly nestačí v dot-bracket formáte jeden druh zátvoriek. Na obrázku 4.1 je jednoduchá RNA s jedným pseudouzlom, ktorú chceme zapísať v dot-bracket formáte. Sekvencia na obrázku začína šiestimi nespárovanými nukleotidmi, ktoré sú reprezentované znakom “.”. Nasleduje šesť nukleotidov, ktoré tvoria prvú polovicu páru a pre ich zápis je použitá zátvorka “(”. Sekvencia pokračuje ďalšími nespárovanými nukleotidmi, až kým nenarazíme na tri spárované. Ide o prvé nukleotidy v páre a preto by mali byť označené znakom “(”. Problém nastane hneď pri ďalšom nukleotide, ktorý je označený zátvorkou “)”. Takto zapísaná RNA je úplne iná štruktúra akú sme chceli vyjadriť, pretože ľavá a pravá zátvorka s rovnakou úrovňou zanorenia reprezentujú pár. Z tohto dôvodu sa pre zápis pseudouzlov typicky používajú hranaté zátvorky. Obrázky 4.2 a 4.3 ukazujú rozdiel vo význame rôznych zápisov.

Pôvodná verzia Traveleru sa vedela s pseudouzľami na vstupe vysporiadať, neriešila však ich vizualizáciu. Pri parsovaní Fasta súborov obsahujúcich popis párovania v dot-bracket formáte program zahodil informáciu o pseudouzľoch a namiesto páru nukleotidov si zapamätal dva nespárované nukleotidy. Tento prístup umožňuje implementovať vizualizáciu pseudouzľov bez toho aby bolo potrebné meniť datovú štruktúru, ktorá reprezentuje RNA. Traveler narába s RNA ako so stromami. V tomto strome si môžeme predstaviť pseudouzol ako hranu medzi dvoma listami. Pre vizualizáciu pseudouzľov je teda potrebné aby program evidoval dvojice listov, ktoré tvoria jednotlivé pseudouzly a po vykreslení celej sekundárnej štruktúry znázorniť párovanie medzi nukleotidmi. Obrázok 4.4 bol vygenerovaný nástrojom Traveler po doplnení funkcionality spracovávaní pseudouzľov.



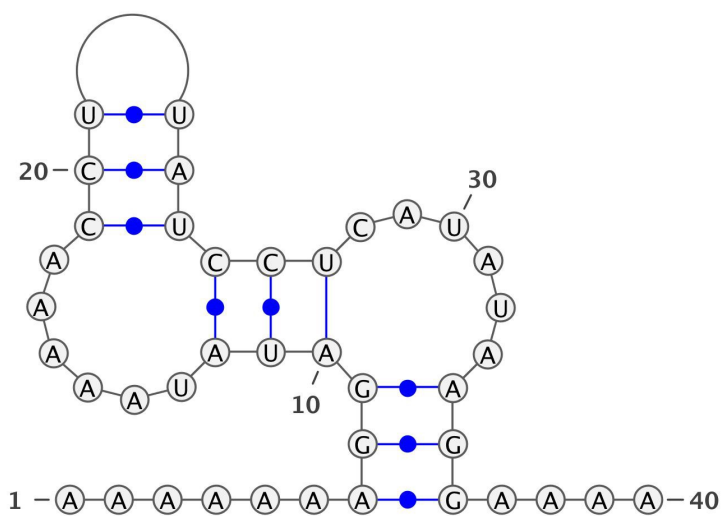
Produced with jViz.Rna - <http://jviz.research.ist.slu.se>

Obrázok 4.1: Príklad RNA pseudouzľu (Wiese a kol., 2005)



.....(((((((.....[[[D]]))))).....]]]....

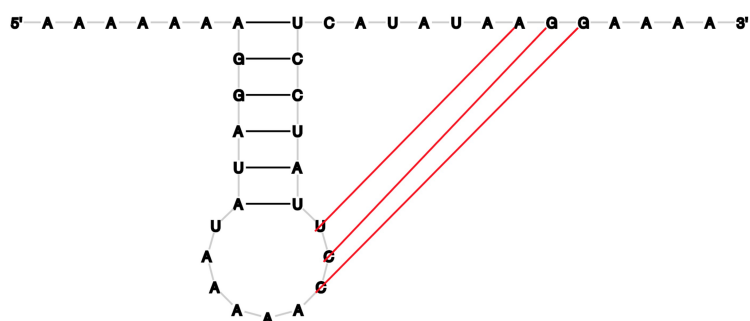
Obrázok 4.2: Korektný zápis pseudouzlov



.....(((((((.....(((())...))....)))....

Obrázok 4.3: Pokus o zápis pseudouzlov bez použitia hranatých zátvoriek. Vzniká úplne iná štruktúra.





Obrázok 4.4: Vizualizácia pseudouzlu pomocou nástroja Traveler 2.0

## 4.2 Medziformát pre obrázky

Formát s jednoduchou štruktúrou príde Traveleru vhod hlavne na vstupe, pri zadávaní šablón. Pôvodne podporované formáty VARNA a CRW sú rozšírené medzi nástrojmi pre analýzu sekundárnej štruktúry RNA. Okrem informácií, ktoré sú relevantné pre vizualizáciu (pozície nukleotidov) ale obsahujú veľké množstvo informácií, ktoré program nepotrebuje. Cieľom pri vytváraní medziformátu bolo poskytnúť možnosť previesť zdrojové obrázky do jednoduchšej formy, ktorej bude Traveler rozumieť. Ďalším problémom je, že existuje veľký počet spôsobov ako zapísať obrázok v pôvodne podporovaných formátoch. Traveler pri parsovaní vstupu predpokladá obrázky vytvorené nástrojom Varna (SVG) resp. CRW (PostScript). Pri inom zápise nebude schopný správne prečítať šablónu.

Čo sa týka štruktúry súborov dodržiavajúcich nový medziformát, zvažovali sme obyčajný textový súbor, ktorého jeden riadok bude zachytávať informáciu o jednom nukleotide. Každý riadok súboru by teda obsahoval dve čísla reprezentujúce súradnice nukleotidu a jedno z písmen A, C, G, U podľa typu nukleotidu. Táto myšlienka bola zavrhnutá, pretože pri pohľade na súbor nie je hneď jasné, čo riadky reprezentujú. Namiesto tohto, až príliš zjednodušeného, formátu sme sa nakoniec

rozhodli reprezentovať rozloženie nukleotidov pomocou XML. Toto riešenie spĺňa požiadavku na jednoduchosť formátu a taktiež je vďaka XML elementom jasné akú informáciu súbor uchováva. Ďalšou výhodou, ktorú tento formát prináša je jednoduché editovanie šablóny, či už ručne alebo niektorým z nástrojov pre prácu s XML DOM.

Pre zachytenie šablóny sú potrebné dva XML elementy:

- **Structure** je element bez atribútov, ktorý obsahuje neobmedzený počet elementov point. Vymedzuje začiatok a koniec štruktúry.
- **Point** element nesie vo svojich atribútoch informáciu o jednom nukleotide. Má tri atribúty: x, y, b. Atribúty x a y reprezentujú x-ovú, resp. y-ovú súradnicu nukleotidu, atribút b (skratka z „base“) nesie informáciu o type nukleotidu.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="structure">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="point" maxOccurs="unbounded" minOccurs="0">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:string">
                <xs:attribute type="xs:float" name="x" use="optional"/>
                <xs:attribute type="xs:float" name="y" use="optional"/>
                <xs:attribute type="xs:string" name="b" use="optional"/>
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Obrázok 4.5: XML Schema pre medziformát

### 4.3 Algoritmus

Problém s algoritmom spočíval v tom, že program zvolil neoptimálnu postupnosť operácií pri prevode šablóny na cieľovú štruktúru. Tento jav nastáva v prípade, že štruktúry sú príliš odlišné v jednej oblasti a takmer zhodné vo zvyšku. Navyše miesto, ktorým sa štruktúry líšia je relatívne malé v pomere k veľkosti celej

štruktúry. Po prevedení štruktúr na stromy program nájde mapovanie, ktoré má najnižšiu cenu ale vedie na nesprávne vykreslenie štruktúry. Na odstránenie tohto problému potrebujeme mapovanie, ktoré nebude globálne optimálne ale bude uprednostňovať lokálne skóre. Zvažovali sme dva spôsoby, ktorými by sme sa mohli dostať k požadovanému výsledku:

- úprava TED algoritmu
- rozdelenie stromov na podstromy a ich následné spojenie

Siahnuť rovno do algoritmu pre výpočet stromovej editačnej vzdialenosti, by prinieslo elegantnejšie riešenie a menší zásah do častí kódu, ktoré s algoritmom nemajú nič spoločné. Pri pokusoch o úpravu algoritmu sa ale ukázalo, že tento prístup je oveľa komplikovanejší ako sme pôvodne prepokladali. Snaha donútiť Traveler aby uprednostňoval pri výbere mapovania lokálne skóre, spočívala v zmene váh jednotlivých operácií v závislosti na hĺbke vrcholov v strome. Operácie, ktoré menili vrcholy vo väčšej hĺbke, mali väčšiu váhu. Táto zmena však nepriniesla požadovaný efekt, práve naopak - vizualizácie boli nepoužiteľné. Keďže sme zasiahli priamo do výpočtu, program nebol schopný vizualizovať štruktúry, ktoré mu predtým nerobili problém.

Po neúspešnom pokuse o zásah do RTED, sme sa rozhodli, že namiesto toho rozdelíme štruktúru na menšie časti a pustíme na ne neupravený algoritmus RTED. Postupujeme nasledovne:

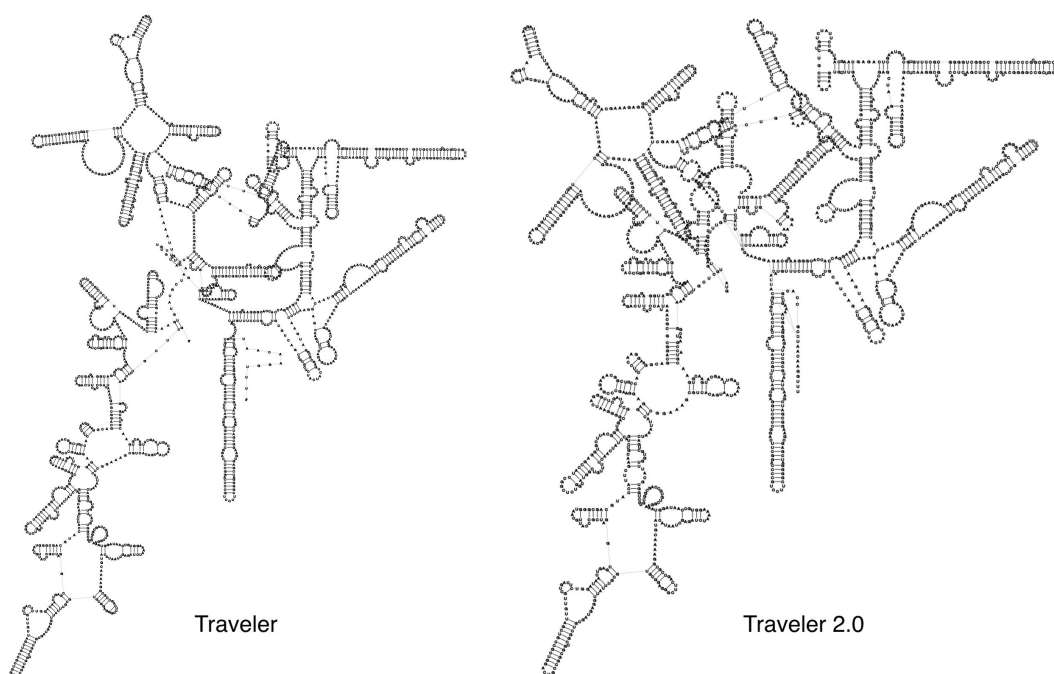
- rozdělíme stromové reprezentácie sekundárnych štruktúr na podstromy na prvej úrovni, t.j. každý zo synov koreňa je teraz koreňom samostatného stromu,
- spočítame TED pre všetky dvojice, kde jeden strom vyberieme z lesa cieľovej štruktúry a druhý strom z lesa šablóny,
- vyberieme dvojice s najnižšou cenou mapovaní,
- vo všetkých dvojiciach samostatne vykonáme prevod šablóny na cieľovú štruktúru,
- po transformácii všetkých podstromov ich znova pripojíme na spoločný koreň,
- takto skonštruovaný strom vykreslíme.

Pseudokód algoritmu:

```
template_forrest := forrest(template)
target_forrest := forrest(target)
pairs, result

FOREACH x IN target_forrest BEGIN
  all_mappings
  FOREACH y in template_forrest BEGIN
    (x,y) = ted(x,y)
    ADD (x,y) TO all_mappings
  END
  ADD min(all_mappings) TO pairs
END

FOREACH (x,y) IN pairs BEGIN
  transform y to x
  ADD y TO result
END
```

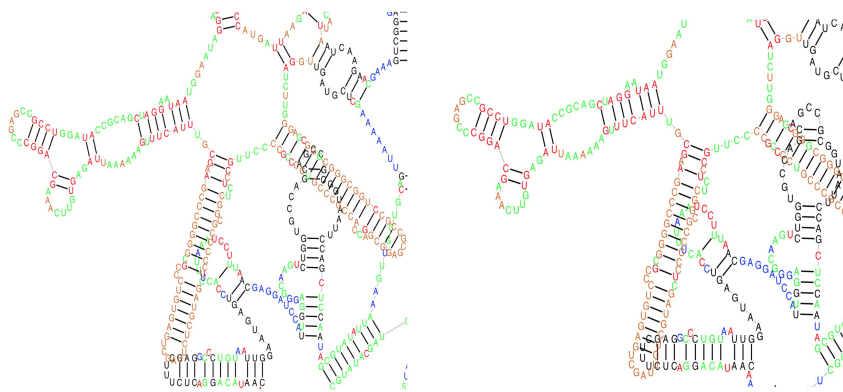


Obrázok 4.6: Porovnanie vykreslenia pôvodným algoritmom a jeho upravenou verziou

Tento prístup produkuje rovnaké vykreslenia ako pôvodný program v prípadoch, kedy nenastáva problém s mapovaním. V situáciách, v ktorých Traveler vyrobil nevhodné mapovanie, sme touto zmenou výrazne zlepšili kvalitu vykreslenia sekundárnej štruktúry. Jednotlivé podstromy sú vykreslené omnoho kompaktnejšie a celé vykreslenie je prehľadnejšie. Aj s použitím rozloženia stromov sa ale vo výslednej vizualizácii vyskytujú prekryvy. Tie sú však väčšinou spôsobené tým, že cieľová štruktúra a šablóna sú natoľko odlišné, že vykreslenie šablóny je nevhodné

pre cieľovú štruktúru. Podstromy sú správne namapované a vykreslené ale ich umiestnenie zadané šablónou môže spôsobiť prekryv.

Pri testovaní upraveného algoritmu sme zaznamenali mierne zvýšenie počtu prekryvov na štruktúrach, ktoré neboli pre Traveler problematické. Dôvodom vzniku nových prekryvov je posun niektorých vetiev štruktúry, čo je následkom rozloženia a následného spojenia jej stromovej reprezentácie. Pri postprocessingu program prechádza celú štruktúru a prepočítava súradnice vrcholov aby minimalizoval zmenu šablóny. Tento výpočet je ovplyvnený mapovaním, ktoré je odlišné pri rozložení štruktúry na menšie časti. Prekryvy v tomto prípade ale nijako neovplyvňujú kvalitu obrázku. Obrázok 4.7 ukazuje príklad tejto situácie.

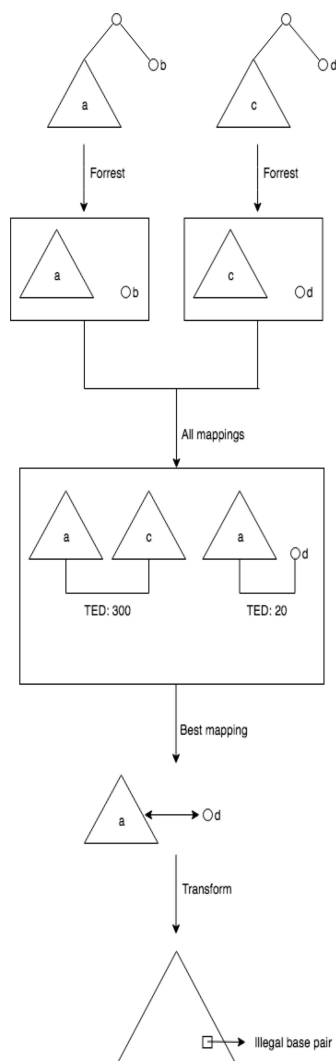


Obrázok 4.7: Prekryvy spôsobené posunutím vetvy. Vľavo Traveler, vpravo Traveler 2.0

Pri rozdelení stromu na les sme museli špeciálne ošetriť listy, ktoré boli bezprostrednými potomkami koreňa. List sa vždy namapuje na list, pretože nie je potrebná žiadna operácia okrem premenovania vrcholu. Počty takýchto listov v štruktúrach môžeme rozdeliť na 3 prípady:

- ak má cieľová štruktúra rovnaký počet listov ako šablóna, je všetko v poriadku, listy sa namapujú 1 k 1,
- ak má cieľová štruktúra menej listov ako šablóna, je všetko v poriadku, prebytočné listy budú zmazané (v rámci postprocessingu sa upraví layout),
- ak má cieľová štruktúra viac listov ako šablóna, niektoré listy sa nebudú mať na čo namapovať. Túto situáciu sme vyriešili dotatočným priradením súradníc listom, ktoré neboli namapované.

Ďalším obmedzením, ktoré sme museli zakomponovať do mapovania dvojíc podstromov je zákaz mapovania podstromu, ktorý obsahuje aspoň dva vrcholy na list. Z hľadiska stromovej editačnej vzdialenosti je táto situácia v poriadku. V kontexte sekundárnej štruktúry RNA ale Traveler v tomto prípade vytvoril nezmyselné dvojice vrcholov, napríklad vrchol s označením GC-A. Dvojica GC sama o sebe reprezentuje pár nukleotidov a nemôže tvoriť ďalší pár s nukleotidom A. Ak je šablóna dostatočne podobná cieľovej štruktúre táto situácia nenastane. V prípade, kvôli ktorému bola potrebná úprava algoritmu tento jav nastáva počas výpočtu niekoľko krát. Štruktúry sú natoľko odlišné, že niektoré podstromy majú bližšie k listu, ako k iným podstromom. Rozdiel v cenách mapovania sa pohybuje v malých stovkách.



Obrázok 4.8: Vznik zakázaného páru vrcholov

## 5 Experiment

Pre automatický výber šablóny webovou aplikáciou sme sa rozhodli použiť nástroj RNAdistance (Lorenz a kol., 2011). To, že RNAdistance vyberá správnu šablónu sme overili experimentom popísaným nižšie.

### 5.1 Popis experimentu

Pracovali sme s dvoma skupinami sekundárnych štruktúr. Prvá skupina boli cieľové štruktúry, ku ktorým sme chceli vybrať šablónu. Druhá skupina, ktorá obsahovala len dve štruktúry, boli samotné šablóny. Štruktúry v prvej skupine pochádzali z rodiny chordata a diptera, pričom pre každú zo skupín je optimálna jedna z dvoch šablón. Nástroj RNAdistance sme nechali spočítať stromovú editačnú vzdialenosť pre všetky štruktúry a všetky šablóny. Čím menšie číslo sa nachádza na výstupe programu, tým podobnejšie sú porovnávané štruktúry.

### 5.2 Dáta

Ako šablóny sme použili tieto štruktúry:

- vlnna muška (*drosophila melanogaster*)
- človek (*homo sapiens*)

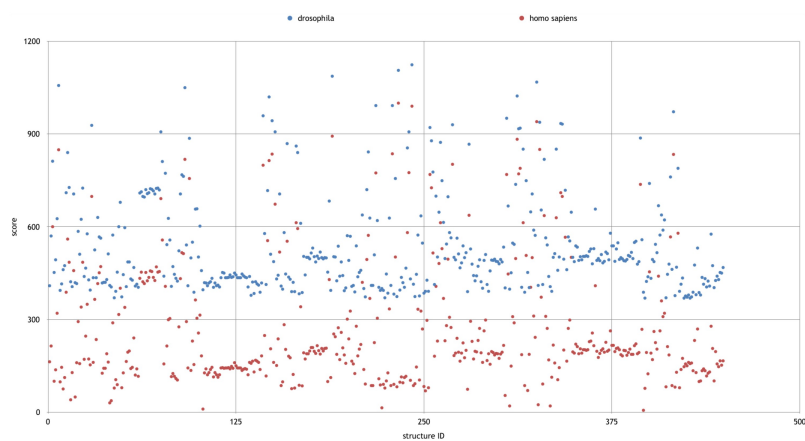
Cieľové štruktúry boli rozdelené na:

- dvojkrídlovce (diptera, 472 štruktúr)
- strunatce (chordata, 450 štruktúr)

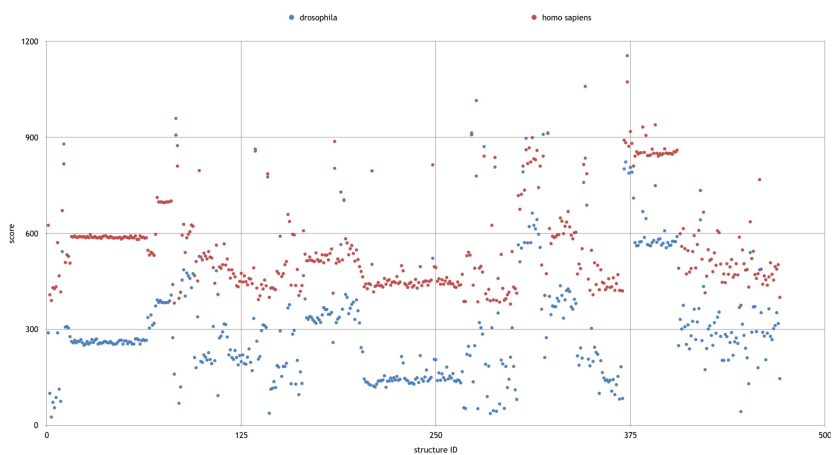
Správnou šablónou pre dvojkrídlovce bola vlnna muška, pre strunatce človek.

### 5.3 Výsledky

Zaujímalo nás v akom počte prípadov vyberie RNAdistance správnu šablónu. Pre strunatce program vždy zvolil správnu šablónu (Obrázok 5.1). Pre dvojkrídlovce sa mýlil v desiatich prípadoch (Obrázok 5.2). Grafy na obrázkoch 5.1 a 5.2 popisujú výsledky experimentu. Os x reprezentuje štruktúry, pre ktoré sme vyberali šablónu, na osi y sa nachádza skóre testovaných šablón.



Obrázok 5.1: Výsledky výberu šablóny pre strunatce



Obrázok 5.2: Výsledky výberu šablóny pre dvojkrídlovce



	vínna muška	človek
AVG	535,48	260,52
MIN	369	6
MAX	1124	1000

Tabuľka 5.1: Štatistika výberu šablón pre strunatce

	vínna muška	človek
AVG	310.76	558
MIN	26	375
MAX	1155	1073

Tabuľka 5.2: Štatistika výberu šablón pre dvojkrídlovce

## Záver

Cieľom práce bolo odstrániť nedostatky nástroja Traveler, ktoré bránili jeho jednoduchému používaniu v praxi. Najväčším nedostatkom bolo komplikované ovládanie programu, čo sme vyriešili vytvorením užívateľského rozhrania v podobe webovej aplikácie. Okrem nadstavby nad samotným programom je aplikácia schopná sama vybrať vhodnú šablónu. Táto funkcia je naimplementovaná pomocou nástroja RNAdistance, ktorý rovnako ako Traveler pracuje so stromovou reprezentáciou štruktúr. Experimentálne sme overili, že RNAdistance zvolí správnu šablónu.

Program sme rozšírili o schopnosť spracovávať pseudouzly. Ich vizualizácia pomocou Traveleru zodpovedá štandardnému vykresleniu pomocou nástroja VARNA. Vytvorili sme medziformát, pomocou ktorého môžu byť reprezentované šablóny. Šablóny v medziformáte majú jednoznačnú formu, ktorú je program vždy schopný rozparsovať. Problém s algoritmom sme zmiernili tým, že pred začiatkom výpočtu mapovania, rozdelíme obe štruktúry na menšie časti. Pre dvojice spočítame vlastné mapovanie, ktoré je lokálne presnejšie ako mapovanie celých štruktúr. Následne na základe mapovania upravíme časti šablóny a spojíme ich dokopy.

Možným vylepšením súčasného stavu programu je pridať schopnosť predchádzať prekryvom. To sa dá dosiahnuť predĺžením hrán v prípade kolízie. Je tým možné predísť prekryvom, ktoré vyplývajú z použitia nevhodnej šablóny.

## Zoznam použitej literatúry

ALBERTS, B., JOHNSON, A., LEWIS, J., RAFF, M., ROBERTS, K., WLATER, P. (2002). *Molecular biology of the cell, 4th edition*. Garland Science, USA. ISBN-10:0-8153-3218-1.

Bootstrap 4.1 (2018), URL: <https://getbootstrap.com/>. Accessed: 2018-04-21.

BYUN, Y., HAN, K. (2009). PseudoViewer3: generating planar drawings of large-scale RNA structures with pseudoknots. *Bioinformatics*, **25**, 1435-1437.

BYUN, Y., HAN, K. (2006). PseudoViewer: web application and web service for visualising pseudoknots and secondary structures. *Nucleic Acids Research*, **34**, 416-422.

CANNONE, J., SUBRAMANIAN, S., SCHNARE, M., COLLETT, J., D'SOUZA, L., DU, Y., FENG, B., LIN, N., MADABUSI, L., MULLER, K., PANDE, N., SHANG, Z., YU, N. A GUTELL, R. (2002). The comparative RNA web (CRW) site: an online database of comparative sequence and structure information for ribosomal, intron, and other RNAs: Correction. *BMC Bioinformatics*, **3**(1), 15+. ISSN 1471-2105. doi: 10.1186/1471-2105-3-15. URL: <http://dx.doi.org/10.1186/1471-2105-3-15>.

CHHEDA, N., GUPTA, M. (2014). *RNA as a Permutation*.

DARTY, K., DENISE, A., PONTY, Y. (2009). VARNA: Interactive drawing and editing of the RNA secondary structure. *Bioinformatics*, **25**(15), 1974-1975.

ELIAS, R., HOKSZA, D. (2017). TRAVeLer: a tool for template-based RNA secondary structure visualization. *BMC bioinformatics*, **18**(1), 487.

ELIÁS, R., HOKSZA, D. (2016). Rna secondary structure visualization using tree edit distance. *International Journal of Bioscience, Biochemistry and Bioinformatics*, **6**(1), 9.

HOFACKER, I.L., FONTANA, W., STADLER, P.F., BONHOEFFER, S., TACKER, M., SCHUSTER, P. (1994). Fast Folding and Comparison of RNA Secondary Structures. *Monatshefte f. Chemie*, **125**, 167-188.

HOFACKER, I.L., STADLER, P.F. *RNA Secondary Structures*. URL: <http://www.bioinf.uni-leipzig.de/Leere/SS15/Bioinf2/lengauer.pdf>. Accessed: 2018-05-10.

KOONIN, E.V. (2005). Orthologs, paralog, and evolutionary genomics. *Annual Review of Genetics*, **39**, 309-338.

LORENZ, R., BERNHART, S.H., HOENER, C., TAFER, H., FLAMM, C., STADLER, P.F., HOFACKER, I.L. (2011). ViennaRNA Package 2.0. *Algorithms for Molecular Biology*, **6**(26).

LORENZ, R., HOFACKER, I.L., STADLER, P.F. (2016). RNA folding with hard and soft constraints. *Algorithms for Molecular Biology*, **11**(1), 1-1.

PAWLIK, M., AUGSTEN, N. (2011). RTED: a robust algorithm for the tree edit distance. *Proceedings of the VLDB Endowment*, **5**(4), 334-345.

PONTY, Y., LECLERC, F. (2015). Drawing and Editing the Secondary Structure(s) of RNA. *RNA Bioinformatics*, **1269**, 63-100.

TAI, K.-C. (1979). The tree-to-tree correction problem. *J. ACM*.

WIESE, K.C., GLEN, E., VASEDUVAN, A. (2005). jViz.Rna java tool for RNA secondary structure visualization. *IEEE Transactions on NanoBioscience*, **4**(3), 212-218.

YANG, H., JOSSINET, F., LEONTIS, N., CHEN, L., WESTBROOK, J., BERMAN, H.M., WESTHOF, E. (2003). Tools for the automatic identification and classification of RNA base pairs. *Nucleic Acids Research*, **31**(13), 3450-3460.

## Zoznam obrázkov

Obrázok 1.1: Príklad sekundárnej štruktúry RNA.....	6
Obrázok 1.2: Príklad terciárnej štruktúry RNA (PDB ID: 3J7A).....	7
Obrázok 1.3: Motívy sekundárnej štruktúry RNA (Chheda a Gupta, 2014).....	8
Obrázok 2.1: Príklady zobrazenia sekundárnej štruktúry RNA, zľava: rovinný graf, kruhové rozmiestnenie a lineárne rozmiestnenie.....	11
Obrázok 2.2: Vizualizácia sekundárnej štruktúry RNA človeka pomocou nástroja Varna (Darty a kol., 2009).....	12
Obrázok 2.3: Vizualizácia sekundárnej štruktúry Amblyospora bracteata 18s rRNA pomocou nástroja Traveler (použitá šablóna: 18S human rRNA).....	13
Obrázok 2.4: Očakávané zobrazenie sekundárnej štruktúry Amblyospora bracteata 18s rRNA (Cannone a kol, 2002).....	14
Obrázok 2.5: Úprava šablóny pri vkladaní nukleotidov (Eliáš a Hoksza, 2016).....	15
Obrázok 2.6: Príklad problému s mapovaním.....	17
Obrázok 2.7: Príklad spustenia programu.....	19
Obrázok 3.1: Frontend.....	22
Obrázok 3.2: Backend.....	23
Obrázok 3.3: Formulár pre zadávanie vstupných dát.....	26
Obrázok 3.4: Prezentácia výsledku pomocou webovej aplikácie.....	27
Obrázok 4.1: Príklad RNA pseudouzlu (Wiese a kol., 2005).....	31
Obrázok 4.2: Korektný zápis pseudouzlov.....	32
Obrázok 4.3: Pokus o zápis pseudouzlov bez použitia hranatých zátvoriek. Vzniká úplne iná štruktúra.....	32
Obrázok 4.4: Vizualizácia pseudouzlu pomocou nástroja Traveler 2.0.....	33
Obrázok 4.5: XML Schema pre medziformát.....	34
Obrázok 4.6: Porovnanie vykreslenia pôvodným algoritmom a jeho upravenou verziou.....	36
Obrázok 4.7: Prekryvy spôsobené posunutím vetvy. Vľavo Traveler, vpravo Traveler 2.0.....	37
Obrázok 4.8: Vznik zakázaného páru vrcholov.....	38
Obrázok 5.1: Výsledky výberu šablóny pre strunatce.....	40
Obrázok 5.2: Výsledky výberu šablóny pre dvojkrídlovce.....	40

## **Zoznam tabuliek**

Tabuľka 5.1: Štatistika výberu šablón pre strunatce.....	41
Tabuľka 5.2: Štatistika výberu šablón pre dvojkrídlovce.....	41